

S I N C L A I R

Every month £1.45 March 1989



WORLD

DEALING WITH DISC DRIVES

Advice for aspiring drivers

PROGRAM OF THE MONTH

Graph Generator

SEARCH AND DELIVER

A more efficient
approach to Archive

REVIEWS

Home Banker
Biograph

TROUBLESHOOTER

4 6 7 8 13 14 18 22 CHECK PHONE NO.
26 28 34 DISC DRIVE 40 45 BACK COVER
OUT SIDE



Editor

Helen Armstrong

Chief Sub Editor

Harold Mayes MBE

Production Manager

Nick Fry

Designer

Chris Winch

Advertising Sales

Judi Gallon

Robert Cole

Magazine Services

Sheila Baker

Advertising Production

Michelle Evans

James Pyle

Managing Editor

Brendon Gore

Publisher

Perry Trevers

Publishing and

Commercial Director

Paul Coster

Financial Director

Brendan McGrath

Chief Executive

Richard Hease

Microdrive Exchange 089 283
 4783/2952 (2 lines) TIL

Sinclair QL World
Greencoat House
Francis Street
London SW1 1DG
Telephone 01-834 1717
Fax 01-828 0270
Telex 9419564 FOCUS G
ISSN 026806X

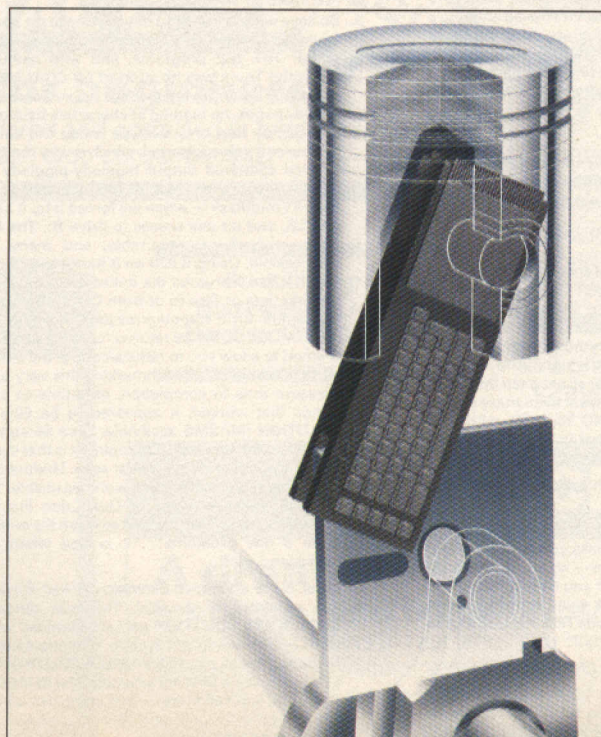
Unfortunately, we are no longer able to answer enquiries made by telephone. If you have any comments or difficulties, please write to The Editor, Open Channel, Trouble Shooter, or Psion Solutions. We will do our best to deal with your problem in the magazine, though we cannot guarantee individual replies. Back issues are available from the publisher price £2 U.K., £2.75 Europe. Overseas rates on request. Please telephone 089 283 4783 to check availability.

Published by Focus Magazines Ltd., London. Distributed by Quadrant Publishing Services, Sutton. Subscription information from: TIL, PO Box 74, Paddock Wood, Tonbridge, Kent TN12 6DW. £15 U.K. £30 Surface mail Europe and the rest of the world. Add £5 for air mail + £10 overseas. Typesetting by Adtec Typographics, Stanford-le-Hope, Essex. Tel: (0375) 360967. Printing by Southernprint Ltd. ©COPYRIGHT SINCLAIR QL WORLD — 1989

CONTENTS

■ ■ MARCH 1989

- 9 **QL SCENE** ● Text '87 update
- 10 **OPEN CHANNEL** ● Speed wars
- 14 **SHOW REPORTS** ● Quanta and the Microfair
- 16 **CAVEAT EMPTOR** ● A realistic approach to purchasing
- 18 **TROUBLESHOOTER** ● Around the scene
- 24 **SOFTWARE FILE: BIOGRAPH** ● Ruling the biowaves
- 26 **SOFTWARE FILE: HOME BANKER** ● DJW Software counts
- 28 **A BETTER SEARCH** ● Archive with greater efficiency
- 34 **DISC DRIVE DIRECTIONS** ● What goes on inside
- 40 **PROGRAM OF THE MONTH** ● Graph Plotter by John Banks
- 46 **MICRODRIVE EXCHANGE** ● Progs for peanuts



NEXT MONTH

SOLO FOR TWO HANDS

The promised keyboard review is here at last — but only one keyboard, from ABC Electronik

DIY TOOLKIT

Goodwin is really excited about this one. It makes bad-tempered software multi-task on the QL, whether it likes it or not.

QL

SCENE

Sharp Deal

Sharp's of Mechanicsville has bought all the remaining American QLs from A+ Computer in the States and now has more than 600 QLs in stock. They are offered for £89 including surface postage to European countries including the U.K., with a surcharge of £15 for air mail delivery.

For further information contact Sharp's, Box 326, Mechanicsville, VA 32111, USA. Tel: 804 730 9697.

FUN AND GAMES

Computer graphics designer Nigel Holder is on the verge of publishing a series of low-cost games for the QL. The games will all cost less than £10 and will be written either in machine code or in a combination of Turbo-compiled SuperBasic and machine code sprite routines.

The first game available will be *International Cricket*, with alternative 256K and 512K versions costing £4.99, or £3.49 if the buyer supplies the micro-cassette. Prices include post and packing. Planned titles include a 3D golf game and an "alternate reality adventure."

More information from Nigel Holder, 25 Beightons Walk, Healy Gardens, Rochdale, Lancashire OL12 6EA.

NEW TEXT, '89

Software 87, publisher of word processor *Text 87*, has produced a version 2 of the program with many new features,

as well as three accessory programs, *Fountex88*, *Founded89* and *2488*, a set of text-mode printer drivers for Epson and

NEC 24-pin printers. New features provided by the V2 include screen justification, screen ruler for the cursor line, WYSIWYG display of typefaces by multiple graphics fonts, plus the name of the typefaces in use displayed, 10 typeface options for quick insertion in the text, five extra screen display fonts supplied, and an extended 80-page manual with examples.

Further upgrades to the program planned in 1989 include printer drives for Hewlett-Packard laserJet and DeskJet printers and subsequently for PostScript printers. A German version of *Text87* is being developed and may be followed by other European language versions.

Fountex88 is a printer driver for graphic bit-image corresponding with the screen display fonts used with *Text87*. *Fountex88* provides 30 extra fonts between eight and 76 pixels high and allows up to 32 fonts to be loaded and mixed on-screen. *Founded88* allows the creation or editing of characters up to 84 x 96 pixels and captures images saved from QL paint programs. *Fountex88* has many other features and costs £25. *Founded89* costs £15.

More information from Software87, 33 Savernake Road, London NW3

101-Swiss_light ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 102-Swiss_bold ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 103-Typewriter16 ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 104-Typewriter24 ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 105-Antiqua ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 106-Antiqua2 ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 107-Antiqua20 ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 108-COMPACTA ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 109-COMPACTA OUTLINE ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 110-SUPERSTAR ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 111-Oldenglish ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 112-Gothik ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 113-Broadway ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 114-Corinthian ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 115-Neon ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 116-Data ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 117-Digitol ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 118-Boxed ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 119-Shadow ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 120-Chicago ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 121-Chicago Italic ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 122-Egyptienne ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 123-Grotesque ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 124-Polara ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 125-Modern ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 126-Avari Gorde ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz
 127-Roma ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz

Type-styles from the new *Text87* update.

'Alternative' Show in April

The organisers of the Alternative Micro Show have announced a London version to be held at the New Horticultural Hall, the venue used by the regular ZX Microfairs, on Saturday, April 1.

"Our definition of 'alternative' is not Amiga, ST or IBM

and its compatibles," says organiser Graham Bettany of Emsoft, which hopes to attract QL suppliers and users among those of many other established computers. User groups are offered the opportunity to exhibit at greatly-reduced rates "to encourage the exchange of

computer ideas." Not all user groups are delighted by the Emsoft initiative.

The National Dragon User Group, which is planning its own London show in April, fears that a larger competitor, fuelled by the custom of several computer communities, could

"wreck what is left of the Dragon scene." It may prove difficult to establish a multi-machine show without clashing with one or other of the shows promoted for individual machines by dealers or user groups.

Further information from Emsoft Ltd, Popular Lane, Copdock, Ipswich IP2 0BA. Tel: 0473 690729.

OPEN CHANNEL

Open Channel is where you have the opportunity to voice your opinions in *Sinclair QL World*. Whether you want to ask for help with a technical problem, provide somebody

with the answer, or just sound off about something which bothers you, write to: Open Channel, Sinclair QL World, Greencoat House, Francis Street, London SW1P 1DG.

Happy

Further to the letter in the October issue of *QL World* from W. G. Johns about the user group he was starting, I thought I would write to tell you how it is developing. I was a new user when I bought my first copy of *QL World*. That was 12 months ago; since then I have met Johns many times and spent many happy hours on his QL.

The club has now expanded to about 40 to 50 members from all over the world, with a wide selection of interests, all of which are of interest to myself and many other people who do not understand the jargon of experts talking to

experts. So we try to help in a simple way.

We also have a newsletter which we try to send on a regular basis; it is sent on Microdrive. We managed to get the first issue out in time for Christmas.

Anyone interested in participating in the newsletter, even people who know what they are talking about, will be made welcome.

Mike Kenneally,
QLI,
6, Barnaby Road,
Poynton,
Cheshire,
SK12 1LR.

Silk Strikes

I appear to have offended Gus Chandler in my review of *Light-*

ning. The latest available version of *Speedscreen* was used in all comparisons and tests. The earliest distributed version of *Lightning* was used for the review.

Chandler claims that I made a number of mistakes. The traditional tools of the critic are analysis and comparison. I analysed *Lightning*. I compared it to the previously-published *Speedscreen*. Bearing in mind that my purpose was not to review *Speedscreen* but to use that program as a yardstick for *Lightning*, let me deal with the alleged mistakes.

So far as I know, Chandler does not have a copy of the PDQL SEDIT. With *Speedscreen* installed, SEDIT "froze" when the Block Graphics mode was invoked. If there is a later "fix" I am glad to hear it, but I have not seen it.

I implied, correctly, that different versions of *Speedscreen* were required or, at worst, desirable for each QL-compatible model. For the sake of clarity they are EPROM for a standard 128K machine, disc/cartridge for extended memory-based machines and a ROM version for the Thor. *Lightning* users may transfer freely between Thor and QL. This was not an adverse criticism against either program — merely a relevant item of information of which potential users should be aware. Chandler restates the position and in the same breath attacks me for having mentioned it in the first place.

He, says that I used the "slowest" version of *Speedscreen* for my timings. Unfortunately the *Speedscreen* manual, more concise than that of *Lightning*, does not spell out the various time comparisons; the manual implies that all versions speed operations equally well, not that some should be avoided because

they are slower. Chandler is wrong about why CST selected *Speedscreen* for its ROM accelerator for the Thor when he states that the decision was taken after a "head-to-head comparison".

The *Speedscreen* ROM was in existence many months before even the *Lightning* beta test copy was available. The author of *Speedscreen* is on record as saying that benchmarks are a waste of time and no sure guide to performance.

Chandler states that my timings are contradicted by "independent tests". My tests were both independent and exhaustive. Each test described in my article was repeated three times, timed by stopwatch and an average taken. They were all made independently of the publishers of both products. I stand by them.

Chandler should re-read my article. It was stated clearly that *Lightning* was available only as RAM-based software, equally that ROM-based routines, available only with *Speedscreen*, will run faster than RAM-based software. The recommended retail prices are RAM-based *Speedscreen* £29.95; *Lightning* £29.95; Standard *Speedscreen* £20.

Lightning offers graphics and maths extensions; *Speedscreen* does not. They both offer extra fonts — *Lightning* a greater number than *Speedscreen*. *Lightning* offers one or two extra facilities as described in my article.

All product reviews are inevitably subjective. Where they are based on benchmarks one can draw certain conclusions on a more objective basis. I see no reason to amend anything in the article, except as qualified in or extended by this letter.

John Silk,
PDQL,
Birmingham.

Editor's notebook

I HAVE heard a few horror stories about users who sent money to software suppliers and in some cases never heard of the supplier again. That is the bad news. The good news is that QL users are not the main victims. Other machines in more fashionable, faster-moving markets seem to attract the sharks first.

So what about the advertisements? *QL World* does not accept advertisements from organisations which it believes to be acting suspiciously but, if a company has business problems, we learn only when they begin to affect our readers. Many apparent problems result from unreceived orders, staff holidays or overwork and do not reflect a general state of ill-health.

This is why we pay 'Troubleshooter' Bryan Davies to investigate complaints where readers have tried and failed to get an answer. *QL World* is an independent magazine, with no direct leverage on any company advertising goods or services, other than goodwill, which we try to maintain for everybody's sake.

Please avoid telephoning to threaten us with consumer programmer on television, or recording our replies, which is usually both illegal and pointless. We help where we can.

More than 200 QL users visited the Quanta User Group workshop at its new Northampton venue. The two-day workshop featured presentations by many big names of the QL scene and provided a meeting-place for enthusiasts from all over the world.

It was the first major Quanta function since the group was re-organised in the spring. Previous meetings were held at Swindon, in conjunction with the National Z-80 User Group.

The new location is a community centre a few miles from the M1. It has much more reliable mains power than the Swindon hotel, which was regularly blacked-out when dozens of computers were plugged-in in the meeting hall. This time admission was free to members and several new members were signed.

Kingsthorpe Community Centre is a fascinating building, shared by railway, radio and other clubs in the Northampton area. A wide spiral staircase links the cafeteria and meeting room with the computer room on the floor above; it looks good but poses an obstacle to wheelchair-bound visitors.

The meeting room was packed to overflowing when Tony Tebby spoke on the Saturday afternoon. Much of his audience was standing round the edges of the room, or spilling into the passages outside.

Amid much competitive axe-grinding directed at CST, Tebby announced plans to write a new QL-compatible operating system to run on Atari ST circuit boards. The plan follows the demise of his QLT and Futura designs, which never went into production.

The CST marketing arm, Thor International, was represented by John Silk of PDQL, who discussed Thor XVI production hiccups and demonstrated *Multi-DIScover*, a much-improved version of Dave Walker's disc format conversion program.

Walker gave his own presentation for neophyte assembly language programmers. Mike Lilley introduced QL-PC, his new QL clone, and Tony Firshman went on-line with a collection of QL communications packages.

I was there to discuss ROM

QUANTA USER GROUP WORKSHOP

The first time at Northampton

bugs, DIY Toolkit, as well as answering questions on QL esoterica and also discussing the philosophy underlying my software designs Speedscreen, Turbo and Supercharge.

Upstairs, members had set up much-expanded QL systems. At one point on the Saturday there were 23 QLs in use, along with one Thor XVI, two Atari STs running the Strong Computers *QL Emulator*, and the new QL PC, boxed and in pieces.

Other computers put in an appearance as QL accessories. They included several Psion Organisers and a Sinclair Z-88. There was even a solitary IBM PC clone but that seemed to be switched off all day.

The QL PC is a DIY clone, based on a new circuit-board designed by Quanta member Mike Lilley. The custom chips and board are available from Quanta but it is for you to fill it with a mixture of standard and Sinclair parts.

Despite its name, the QL PC is compatible with the QL but not the IBM PC. It fits in a small instrument box hooked to a standard IBM keyboard. Inside there is 512K of RAM, plus proper clock and Centronics and serial ports.

Buffered

Expansion potential includes four buffered slots for add-on cards, one QL-style connector and sockets for up to 192K of ROM. At the moment the machine runs 3.5in. floppy discs but a Winchester disc interface is in the pipeline.

Elsewhere in the computer room, Walker had a home-made Winchester disc adapter up and running on his QL. Another enthusiast had taken a different route to disc expansion, with no fewer than four Microdrives, three 3.5in. and three 5.25in. drives hooked to one QL.

It was often difficult to identify the Sinclair computer from

the outside. Custom keyboards were much in evidence and several users had re-potted their machines in solidly-made Memotech computer cases.

Former *Quanta* editor Leon Heller was experimenting with a computer-controlled transmitter, broadcasting packet radio data round the building. Normally transmitters interfere with computers but in this case the boot was on the other foot. Heller's main problem was sifting the transmitted signal from the banshee wail of radio interference generated by 30-odd plastic-cased computers running in the same room.

Commercial

The commercial side of the workshop was low-key, as is usual for a Quanta meeting. Firms present included Adman Services, Creative Code-Works, PDQL, QJump, Sector Software and Tony Firshman Services, so software, spares and advice were available in abundance. Working QL circuit boards sold for £40 with chips, or £15 without; broken boards could be bought for £7.50.

Despite minimal publicity, organiser Dennis Briggs estimated the attendance at more than 150 on Saturday, with 40 or 50 new visitors on Sunday. The move to Northampton seems a good one and it looks as if Quanta will need to arrange extra space for its next weekend workshop, scheduled for March 11, 1989.

Quanta is a non-profit-making international group. It runs regular evening meetings in cities all over Britain. Overseas sub-groups meet monthly in Australia, Belgium, Denmark, France, Germany, Norway, Sweden and the U.S.

For further details, contact the secretary, Phil Borman, at 15 Grosvenor Crescent, Grimsby, South Humberside DN32 0QL, or telephone 0472 49850.

The recent ZX Microfair in London found the QL scene as busy as ever, with most *QL World* advertisers in attendance among others. 'New' QLs are still widely available at around £100 each.

New arrivals from Belgium was Progs, touting *The Painter*, a QL graphics package capable of handling 12 screens and 18 fonts at one time. Stepless zooming and inter-screen copying set this apart from earlier pixel-painting packages.

The Painter multi-tasks in a minimum of 256K RAM and works with the Sandy and Q-Jump mice. It costs £32 on 3.5in. disc only, direct from Progs, PB 238, 3000 Leuven 1, Belgium.

Jochen Merz Software from Germany specialises in Q-PTR accessories and suggests that



John Silk of PDQL.

you could use its new £16 Thing and EPROM Manager to put QRAM in ROM. Would not that be QRUM? It also sells a mouse-controlled editor, several games and a pack of Q-PTR utilities through its U.K. agent QJump.

Top hardware news was Miracle Systems' planned Hard Disk unit. It plugs into the QL cartridge port, making it compatible with the best-selling Trump card. You can still plug a 16K ROM cartridge into a duplicate port on the disc box.

Miracle could exhibit only the empty metal case at the fair but by the time you read this it expects to have added a 30 MB Winchester disc drive, IBM-type ST506 controller and QL adapter. The lot should cost

ZX MICROFAIR

Simon Goodwin reports from the New Horticultural Hall



Rob Roy Software in pensive mood.

£399. If you are not afraid of soldering and would like a cheaper QL modification, Jonathan Oakley's CAPS-LED circuit may fit the bill. This £5 project adds a flashing indicator to the QL keytop, so you can tell when you have pressed CAPS LOCK or CTRL-F5.

Oakley has left QJump to work as a £350-a-day consultant but his QL creations are still available from QView at 29 Carnaby Close, Godmanchester, Cambridgeshire.

Programmer Joe Hafke was demonstrating his £20 Income Tax and Home Budgeting package on the PDOL stand, alongside the latest fruits of Chas Dillon — BASIC-C-PORT and XREF 200.

Digital Precision had its usual stall but there was no sign of boss Freddy Vaccha away in India — or the much-mentioned PC Emulator, still being developed at time of

going to press; nonetheless, it was able to offer aggressive discounts on all its old products.

The best bargains were to be had on the QL World stand, where all Microdrive Exchange titles had been reduced to £2 a copy in a special offer.

Microfairs are a focus for user groups as well as commercial firms. The leading group, Quanta, has a working party developing its own hard disc system. Several members have already built their own interfaces — including one controlled, bizarrely, by an Inmos transputer. Quanta is reorganising its library of several megabytes of public domain software, in preparation for the day it can put it all on to one drive.

QL Adventurers Forum was at the fair, selling newsletters and a brilliant new game from Alan Pemberton, the graphic space adventure *Starplod*.

QL SUB booked a stand but was not seen on the day and the German QL Professional User Club was represented on the Ultrasoft stand.

QL SUPERTOOL KIT II by Tony Tebby THE ULTIMATE QL ENHANCEMENT

Over 118 Commands:— Full Screen Editor, Key Define Print Using, Last Line Recall, Altkey, Job Control, File Handling, Default Directories, Extended Network
16k Eprom Cartridge Version @£ 24.15d
Configurable Version on Microdrive @£ 23.00d

MIRACLE SYSTEMS PRODUCTS

QL Trump Card 768k (Toolkit II etc) @£293.25d
QL Trump Card 512k (Toolkit II etc) @£224.25d
QL Trump Card 256k (Toolkit II etc) @£155.25d
QL Expanderam 512k Thru Card £155.25d
QL Expanderam 256k Thru Card @£ 86.25d
QL Modem @£ 48.30d
QL Centronics Printer Interface @£ 28.75d
QL Modaptor @£ 37.95d
QL Midipack @£ 77.05d

QL HARDWARE

Single 3.5" Disc Drive & (Own PSU) @£ 97.75a
Dual 3.5" Disc Drive & (Own PSU) @£188.60a
"Multi Drive 5.25"-3.5" new fully screened upgraded version £213.90a
Q POWER REG. The only real solution to your QL overheating (switched mode power supply run cold) @£ 23.00c
QL Keyboard Membrane @£ 11.50d
QL Keyboard Contact Pad @£ 11.50c
QL Base Computer SM version @£139.15a
QL Base Computer JS version @£149.50a
Care Eprom Cartridges each @£ 8.05c
Eprom 27128 250n/s 16k @£ 5.75c
QL Microdrive Unit @£ 27.60d
ULA Chip ZX8301 @£ 15.64c

MAGNETIC MEDIA

Microdrives (each) @£ 1.98c
3.5" (each) d/s disc @£ 1.61c
3.5" (10 off) d/s discs @£ 13.80c

THE Q CONTROL BOARD

As specified in the Colin Oppie Connexions Series — Please call for details @£ 89.95d
Analogue/digital chip @£ 26.91c
Cross Assembler 8048 or 6502 @£ 18.40c

QL GAMES

AMBITION By Way Ahead @£ 19.95c
Business Orientated Strategy "super Monopoly" @£ 9.20c
QL YAHTZEE @£ 9.20c

TONY TEBBY Software (QJUMP)

GRAM "new" COMPLETE QL MULTITASKING FRONT END A TRULY AMAZING DEVELOPMENT FROM QJUMP POP IN MENUS, FAST RAM DISC, UTILITES, HOTKEYS SCREEN/WINDOW DUMPS, PRINT SPOOLER ETC @£ 29.90d
QIMI QL Internal Mouse Interface. Keeps your Ram port free @£ 39.79c
QIMI + QRAM @£ 63.25c
QIMI + MOUSE @£ 79.35c
QIMI + MOUSE + QRAM @£ 98.90c
QFLP (Micro/P disc interface upgrade) @£ 14.95c
QMON II Microdrive @£ 19.95c
QMED (Medic disc interface upgrade) @£ 14.95c
QPTR Pointer Interface m/drive @£ 34.50c
QPTR Pointer Interface + 3.5" disk @£ 29.90c
QTPY Type/Spell Checker @£ 29.90c
QPAC Desk top (clock, calander, typewriter etc) @£ 21.85d

ZITASOFT Software By Steve Jones

LOCKSMITH copies M/DRIVE — M/DRIVE @£ 11.50c
4MATTER + LOCKSMITH copies M/DRIVE — DISC @£ 23.00c
SHRIVEL memory shrink prog user definable ie 128k or 192k or 256k etc @£ 6.90c
TOOLCHEST utilities to allow the creation of customised mdv doctor pron @£ 11.50c

SIDEWINDER DELUXE amazing PRINT utilise Screen dumps to any size from postage stamp to 20ft banners. Prints sideways, inverts and scales — Mode 4 and Mode 8, Dumps with grey scale shading. Includes across the webbing label making facility @£17.25c

MONITORS (Price including lead)

Philips BM7502 Green Hi-Res @£ 89.93a
Philips CM8833 Colour Med-Res @£271.41a
Philips AV7300 TV/tuner for above @£ 69.00b
Philips TV/Monitor 1010 @£220.80a
Remote TV/Monitor 1210 @£230.00a

READYMADE LEADS

RGB QL to Photo @£ 5.75c
RGB 8-pin DIN @£ 7.13c
RGB 8-pin DIN (Hitachi) @£ 7.13c
RGB 8-pin DIN (Ferguson) @£ 7.13c
RGB 8-pin to SCART (Euro) @£ 11.50c
5-way PCC 24way 'D' (Printer-Ser 1) @£ 9.89c

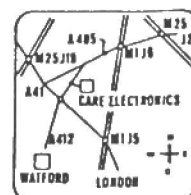
HOW TO ORDER:

ALL PRICES INCLUDE VAT

By Post. Enclose your Cheque/PO made payable to CARE Electronics.
Or use ACCESS/VISA. Allow 7 days for delivery

CARE ELECTRONICS
OPEN
9am-5pm Mon-Thu
9am-4pm Fri-Sat

800 ST ALBANS ROAD,
GARSTON, WATFORD,
HERTS. WD2-6NL.
Tel: 0923-672102



NEW PRODUCTS
New Upgraded
Fully Screened
Q Power Reg

Please add carriage a-£11.50,
b-£5.75 c-£1.15 d-£2.30

QL FIRST FORM Caveat Emptor

Desmond Barry issues some common sense warnings about the purchase of computers and all things connected with them.

Caveat emptor is an old Latin expression meaning let the buyer beware. It is good advice. When I started computing, I took the computer out of the box, fitted it together, plugged it in, and then what?

As it happened, I had something to do. I had been planning a program for some three months so I got on and wrote it. This is not always the case. Many people, on their first contact with a computer, stare at the screen and wonder what they should do next. It is a very worrying feeling.

Then you look at the manual and it seems to be written in a dialect of English of which you have never heard. Many words look familiar but are they not in the wrong order? What is a screendump? As one of my colleagues once said, "a dump is where you put rubbish." What is a file? Like a card file? How can a picture be a file?

How many people have typed E S C instead of hitting the ESCape key? What is a CTRL key? When you first load Quill from a Microdrive, does it not take an eternity? Should it take so long or is there something wrong?

Escapist

Having been reassured that hitting ESC will stop anything, you take a leisurely stroll through the menus, making notes of all the options you do not understand. Then you find there are so many you might as well read the manual. Of course, there are the help pages but they never seem to answer your question, do they? Then you buy a piece of software. It is a utility which quadruples your computing

power, or doubles your speed, or whatever. Why does it not do it?

You buy RAM expansion — what is that, a big sheep? — and find that half your games will not run. You send a letter to the bank manager printed on the printer the shop was giving away with the computer and he is unimpressed with the quality. So you need a new printer.

Some software will not work unless you have expansion RAM, a toolkit with no spanners, or disc drives. If you get as far as multi-tasking, you find that your favourite piece of software does not multi-task, or it interferes with another piece, or it slows the computer or even crashes it.

The cat mistook your master cartridge for a ball of wool and now it will not go into the slot even if you could undo knots. All in all, it is a miracle anyone ever gets into computing. Remember that the QL and Thor are among the more friendly examples. At least you do not get tutorials on directory trees — I always wondered from where telephone books came.

All this and much more frustration is caused by three basic factors — market extravagance; assumptions by programmers that everyone else is also a programmer; and the user not understanding those two points.

We will worry about history, whys and wherefores some other time. For the moment, let us look at the three factors. I am sorry but you will have to live with extravagance. If someone advertises something, you will be told all the good points, none of the bad ones, and any testing will often be on some special case which accentuates the good points. It

is the same for any item advertised, whether it is a computer program, washing powder or dog food.

In most consumer areas you have yardsticks, experience, commonsense and so on to help you decide. Normally you also have a choice between similar items and you have a good idea what you want. The same applies in business. If you need somewhere to store 100 5×3 index cards you do not buy a fireproof, anti-tilt, four-drawer filing cabinet.

As many computer users have discovered, extravagance in the computer market is very expensive and possibly potentially damaging. Most of

"The main reasons for buying computers seems to be; because it is IBM-compatible; it is cheap; it has good graphics; it is popular; or some combinations of all four. It seems rare to buy it because the proper software was available."

the reason is that, to be a discerning buyer, you need to understand what you are buying.

It is a fact that disciplines often have to be mixed and there are few people who mix them effectively. One of the reasons for the dearth of good educational software is that there are few teachers who can program and few programmers who can teach. Unfortunately, everyone thinks they can do both. Consequently, there are very few good educational programs.

Good teamwork can sometimes circumvent this. Remember, however, that a

camel is supposed to be a horse designed by a committee. Do not buy a camel if you need a horse.

Most software is written by programmers with some kind of background in computers. They are not necessarily accountants, artists or typists, so, in many cases, they do not understand what accountants, artists and typists need from a program. On the other hand, the accountants, artists and typists do not program, so they cannot write what they want or possibly even explain everything they need.

Programmers have to work, so they try. The result is often like the original *WordStar*, full of facilities all accessed by CTRL-this and ALT-that, with dots, asterisks and backslashes all over the place. There has probably been more money made from training courses for wordprocessors than from selling the software.

Most programmers probably do not mind this. After all, they make their living out of learning languages, using mnemonics. Why should a user be any different? Believe me, mnemonics are a doddle after direct binary input.

Now, of course, programmers put plenty of icons and little pointers all over the place.

Instead of "yrfile.txt" you have a picture of a piece of paper with the corner folded over and "yrfile.txt" written under it. That makes things much clearer, does it not? As everyone knows, a double click is a single click on the other button and a quadruple click is a single click on the other other button. Obvious, really. I think it is only a matter of time before someone invents a mouse with 84 buttons and calls it a keyboard.

The consumer is the soft pink thing which hits the keyboard at five words per minute and does not need plugging in. The trouble with

consumers is that they do not understand the poor programmer sitting red-eyed at a keyboard at 4am trying to find an ASCII code which has not yet been used after CTRL.

No difference

A programmer will know an elegant piece of code, contrary to popular belief. I have written the odd bit myself. It may make some fundamental difference to a program but very often it will make no practical difference. An average user is interested in getting the job done, not in knowing how it is done. Programmers, I often find, are more interested in the technical side of the job. The consumer should understand this.

Reviews are another potential problem. I have seen some reviews where the item being reviewed was obviously not present at the review. Sometimes it is nothing more than a rehash of company literature, especially when copy dates are short.

The other problem with reviews is that some reviewers

are programmers and tend to look at the product from that rather than the users' angle. Much of the computer world is obsessed with speed. Sometimes it is important. Much of the time it is not, because so much computer work is interactive, i.e., the practical speed of the computer is governed by the printer, the rate of keyboard input, the data transmission speed over a telephone line or whatever.

Many of you probably bought your computer without thinking of it as a consumer item. Talking to people, I have the feeling that the main reasons for buying a computer, apart from the ubiquitous to learn programming, are it is IBM-compatible; it is cheap; it has good graphics; it is popular or some combination of all four.

Right reason

It seems rather rare to find someone who bought it because the proper software was available and yet this should be the number one reason.

The fact that a computer is popular does not mean that the proper software is available. Its popularity could easily be that many people have already made the same mistake. It would be interesting to compare the number of computers sold to the number being used — not counting the ones being used to hold open doors or propping up the wardrobe.

Good graphics is not normally a big plus unless you are into digitising or computer art. This is a skill in itself and if you cannot draw with a pencil you are unlikely to do the Mona Lisa on a computer either.

Cheapness is usually a silly reason for buying anything unless you know what you are doing. IBM compatibility is, I feel, largely a myth. Start with whether you need CGA, EGA or VGA graphics, and an appropriate monitor, of course. Then whether you need 3.5in. or 5.25in. discs. Then whether you can fit standard, short or long expansion cards. Then whether your clone has the correct number of slots of the proper type. Then whether you

have or need 200K of GEM in it — sometimes, it restricts the available RAM enough to prevent certain software going in.

One day, clones will have a multi-tasking operating system which uses only 10 times as much RAM as the QL or Thor — and only four years later some clones now have the reliable 3.5in. discs which have more or less been the standard in our field for two years.

More mistakes

If you want to network on a clone, you pay more. I would not ask too much about programming windows on a clone if I were you — and I have heard people say they were thinking of upgrading to a clone.

Remember that the choice of software of clones is much wider. You may think that is an advantage but if you have made mistakes buying relatively limited choice software on the QL or Thor, think how many more mistakes you could make on a clone. Think how much sophistication you already have.

DE BONDT ELECTRONICS

LV HOEYMISSENSTRAAT 50
2910 MALDEREN — LONDERZEEL, BELGIUM

TEL: 052/33.16.56
FAX: 052/35.8395

Z88 PROGRAMMES

ZECK

Spelling checker with a standard library which can be enlarged very easily by the user. The only limit to the library is the limit to the memory and the storage capacity of the medium.

ZEAD

Prints your logo on your letter. A real drawing programme makes the designing very easy. You can use it with almost every printer.

ZAPHICKS

Converts your numbers of a pipedream document into clear graphs. A whole gamma of two or three-dimensional representatives, easy to use. You can use it with almost every printer.

Name

Address

.....

.....

Payment: Cheque or Credit Transfer to:
GENERALE BANK: 293-0565771-25

Accepted, pst = export price
pst = Pound Sterling
bfr = Belgian Francs

ZERGE

An easy to use, fast and flexible mailmerge.

ZEDIT

Programming in Basic becomes very luxurious with this Basic Screen Editor. Besides the standard possibilities of a Screen Editor, this gives you a lot of utilities which make life much easier. Even the use of a library is supported.

ORDER FORM

ITEM	PRICE	QUANTITY	PRICE
ZECK	PST BFR 5395		
ZEAD	PST BFR 3695		
ZAPHICKS	PST 49 BFR 3995		
ZERGE	PST 47 BFR 3795		
ZEDIT	PST 49 BFR 3995		
TOTAL:			

EXPORT ENQUIRIES WELCOME

T R O U B L E

Bryan Davies scans some of the new software and hardware on the immediate horizon.

Miracle Systems has a hard disc sub-system for the QL. It has been making the running with QL hardware in recent times and it has been left to it to plug the remaining significant gap in the range of add-on units. There have been several hard disc projects through the years and it is difficult for the outsider to see why they all collapsed. Almost any other computers of note have had a hard disc for some time.

There are two points about the Miracle offering which deserve particular praise. One is the projected price of £399, which is appreciably lower than the figure talked about for the last major project to fall by the wayside and should be attractive enough to ensure reasonable sales. The other is the use of a standard 30MB drive unit — the Western Digital brand is quoted — rather than an obsolete 5-20MB type, as was likely with some other projects. This means the drive will be essentially the same as those used on PCs — well-tested, fairly up-to-date and fast.

The step up from Microdrives to discs makes a dramatic improvement in operations and the next step to hard disc is scarcely less of an improvement for the heavy user. When you run systems with and without hard disc side by side you soon realise how great is the speed differential. If that were the only advantage it would be difficult to justify spending £400 but the ability to store all frequently-used files on one drive and to place files in an ordered sub-directory structure will be important to many serious users.

Do not think 30MB is so much space that you will never fill it. I have reduced the spare space on my 32MB unit to about 3MB in six months and will be looking for a larger unit soon. To some extent you can economise on purchases of floppy discs once you have a hard disc but it is all too easy to neglect to make regular back-ups; somehow the ease of operation with hard disc lulls one into thinking back-ups are no longer necessary. Whatever you do, do not fall into this trap — you lose many files if the hard disc directory becomes cor-

rupted. All credit to Miracle for developing this much-needed piece of hardware and I wish it similar success to that it had with Trump Card.

A reader wrote about the comments made in the recent Odd Jottings article concerning modifications which can be made to Trump Card to cure certain problems. The points were made clearly enough but it may be worth giving some history of the Trump again. Early versions gave trouble with Mitsubishi 3.5in. disc drives; the drive lights tended both to be on together, all the time, or they both came on and went off together.

The lights finally were made to operate correctly; in the case of my unit, I had to remove the wire jumper from the underside of the PCB, where the disc drive connector was soldered to the PCB. According to my notes, a length of blue wire was fitted between the second and eighth pins — from the front of PCB — immediately adjacent to the connector. Removing this wire made the lights operate correctly.

Noise

Later PCBs did not have either the lights problem or the wire jumper. The noise produced by Trump in the same drives cannot be eliminated so easily or cheaply. What is needed is a different version of the disc drive controller chip — the WD1772 has to be replaced by a WD1770. This still applies with current units, as no other solution to the noise problem has been found.

Presumably, some users accept the noise and try to ignore it; the drives do not appear to malfunction as a result of it. The other significant problem is locking-up. This can occur for a variety of reasons and the fact that you have lock-ups does not necessarily mean that there is anything wrong with Trump. You should take all the precautions, described many times previously, to get rid of the effects of mains voltage spikes. If you have other add-on units such as a battery back-up clock, you would be well-advised to take them out and do some soak-testing with only Trump fitted before pointing the finger at it as the only possible source of trouble.

The frequency of lock-ups can vary considerably with different QLs and with different chips in the same QL but I have not found a way of eliminating them entirely. Tests are being made with a slightly-modified PCB. At present it appears that the modification reduces and perhaps even gets rid of lock-ups but

not with all QLs. In the meantime, back-up important files frequently.

Another major new arrival, is expected soon. Digital Precision has announced an MS-DOS Emulator. It will be a welcome addition for users who work on both a QL and a PC and should also interest those who want to get some feel for MS-DOS without buying a PC. It has been said by DP chief Freddy Vacha, among others, that producing such an emulator could take several years and the resulting program might not run fast enough to be satisfactory. With DP reputation, and its previous experience producing the CP/M emulator for the QL, the product should be a good one.

Revisions to this very useful program were made towards the end of 1988, and testing has been in progress since then. Several requests for detail changes have been followed; certainly the items on my wish-list all seem to be ticked-off now.

Abacus files can be converted for use in *Flashback*. The code is re-entrant, so several copies of the program can be run alongside each other, each using different turn-on keying. Greater flexibility of cursor movement has been provided. A Replace function has been added to Search. The Index command — cf Archive Order — permits sort order forms — all capitals, then all lower-case, each capital before the corresponding lower-case, and capitals and lower-case in the order they are encountered, for example, ABCabc, or AaBbCc, or aBcDeF. Worthwhile improvements to an already very good program.

Fainthearted

Scarcely had the 1.10 version of *text*⁸⁷ arrived and been looked at than a further revision followed, together with the current versions of *fountex*⁸⁸ and *founted*⁸⁹. This collection of programs is not for the faint-hearted; the concepts involved seem so much out of line with those in the usual run of programs that it is difficult to appreciate for what the author is striving, but the results are worth working for. In fact, it would not be too great a risk to suggest that the screen display and printing capabilities provided are not available in any other word processor program or computer. It would be a pleasure to have this comment demonstrated to be untrue but I will not hold my breath waiting.

Printing must be the operation which

SHOOTER

E M S O L V E D

causes most aggravation to most users. You know before you start that what you see on the screen may be far from what you will get on paper. If you are not ambitious and stick to one character style, perhaps, you have few such problems but step out of line and risk a different character style and you are in for trouble — and plenty of wasted paper.

Handwritten

The correspondence received by *QL World* is evidence of this. It is also remarkable how many letters are handwritten, presumably because the writers find it easier than printing. Quill has the easiest printer setting-up procedure of any WP program I have seen and we all know what a pain it is to produce a printer-data file which gives the desired results. With *text*⁸⁷ you have the facility to make screen and printed page appearance very much alike. You can print the same character styles which appear on the screen, if you make use of *fountext*⁸⁹.

The author is hoping for good sales of *text*⁸⁷ and its associate programs in 1989. It had a slow start, not least because several features were not fully operational in the early versions and the learning process was difficult but the present product is very much better and the instructions are easier to follow.

Equally important, the development process is continuing. For those in doubt about what can be achieved, a look at some of the print samples produced by supplier Software 87 should suffice. The list of complaints I submitted is now down almost to none. At the end of December, Version 2.00 of *text*⁸⁷ was receiving the finishing touches and that will provide several very useful enhancements. There will be a display of the current Ruler settings and an indication of the current typestyle, so that the user will no longer have to try to remember what settings he last made.

It will be possible to set up to 10 typestyles to be selected without recourse to the Type menu; as before, Shift F4 will select one type — the default, small one — but Shift F3 will bring up a menu with the choices 1-9, allowing another nine types to be selected with it.

The type selections will be saved with the printer driver but that feature will not be available at first. On-screen justification is being provided, except for large fonts. Other additions to follow, include another italic font and a double-width font.

The latter is generated by taking a screen shot of the font and making use of the graphics capability of *fountext*⁸⁸.

It has been pointed out that the current version of *QTyp* does not require other QJump products to work with programs such as *The Editor*, *text*⁸⁷ and Quill.

Noticeably absent from the Microfair stalls are copies of Jan Jones' book *QL SuperBasic, The Definitive Handbook*. Are there no stocks of this book hiding anywhere? Judging by the fairly regular references to it in letters, a significant number of a QL users would like a copy. This is the most readable book on the QL I have seen. Would one of our reputable suppliers be prepared to investigate having the book reprinted, possibly against deposits.

Not a direct query this but one which may ring a bell for some DIY users. **Ian Bunting** of Wrexham is working on interfacing the QL to stepper motors, with a view to making a robot. He wants to use the QControl PCB produced by Colin Opie — described in the Connexions series in *QL World* but the design may not be able to cope with the job, as it stands. If anyone has tackled the hardware or software sides of driving a robot from the QL and would like to talk to Bunting, I will pass on the letters.

Brussels

Two suppliers which have not responded to requests for information on readers' problems in recent months are **Schoen** and **GAP**; the readers concerned are **Henri Hulet**, of Brussels, and **T. Calvert** of Oxfordshire, along with a number of other GAP customers.

While the number of "large" QL suppliers is dwindling, one-man suppliers are increasing, it seems. In the long run we may find ourselves with only one major software supplier and one hardware supplier, as some of the familiar names either move out of the QL field completely or devote more time to other machines.

This will not necessarily be a bad thing; I doubt if fears of rocketing prices when there is no choice of supplier will prove to be true. The QL market has grown up as a "cheap" one and it is not likely that users will be prepared to pay PC prices for software or, to a lesser extent, hardware.

The companies doing the most business on the QL scene are not run by people who are in the business simply to make money; they are all genuinely

interested in the QL and dedicated to providing products the users' need.

User groups:

QUANTA

Membership secretary: Phil Borman, 15 Grosvenor Crescent, Grimsby, South Humberside DN32 0QJ.
Tel: 0472 49850.

S.U.B.

PO Box 3, Shildon DL4 2LW
Tel: 0388 450610.

R. Gilbert from Nova Scotia asks why the charges for software are higher to overseas buyers than to U.K. ones and quotes figures to suggest they should be lower. I do not think the reasons can be made fully clear to anyone who has not been in the business of sending goods overseas. The fact is that sending goods overseas is much less simple than sending them in the U.K. and the administrative cost, largely staff time, is much greater. While you can put a personal airmail letter into a local postbox and it will not cost much more than a local first-class letter, commercial packages have to be taken to a post office and forms filled in. If you are VAT-registered it is necessary to have all packages listed on Post Office forms or you are liable to have difficulty with the VAT people, especially if a package is returned, when you can be charged for it.

The reason for some suppliers not deducting anything from the price of software to allow for no VAT being charged is not that they want to pocket the VAT but that they are making an effective increase in the price of the goods to cover some of the extra cost of despatching them.

To suggest, as Gilbert does, that postage charges for sending packages overseas are small is wide of the mark in many cases. To take the extreme, there is no way instruction manuals of the size Digital Precision supplies can be sent for the normal basic postal charge.

The reasons for magazines like *QL World* costing so much more overseas are rather different. The print trade has its own agreements around the world and publishers will not supply magazines direct to readers at a price lower than that charged by their agents in the countries concerned. The cover prices of magazines are normally much higher in foreign currency so, inevitably, the subscription costs will be higher, too.

Information:

BIOgraph
Price £20
Imaqlate Software
42 Albion Street,
Broadstairs,
Kent CT10 1NE

Life has its ups and downs

Do you have days when you seem to be more tired than on other days, or when you sail through the things you have to do, or cannot seem to muster the enthusiasm to get going on something?

Do you attribute those conditions to having a bad day or people just getting on your nerves or just being a day when you should have stayed in bed, blown raspberries at the world and thrown the alarm clock out of the window?

The answer to those questions, and more, could be found in the natural rhythms of the body, known as biorhythms. Biorhythms, it appears, were discovered by European scientists in the early part of the 20th century and were defined as internal body states interacting with each other and influenced by external pressures and the environment. The body rhythms follow a regular pattern throughout life.

They are put into three categories — physical, intellectual and emotional. The physical runs in a 23-day cycle, intellectual in a 33-day cycle and emotional in a 28-day cycle.

Biorhythmic cycles are termed as active or positive during the first half of their duration and inactive or negative during the second half when the energies of a cycle are recharging. The point at which a cycle crosses from active to inactive is known as critical and is the most unstable time of a whole cycle. Research for many years has shown that we are five times more likely to have an accident on a critical day and 11 times more likely to die. The days either side of a critical day are often termed as semi-critical.

These biorhythms can be calculated and shown on a

graph but it takes some complex calculations and understanding to make such a chart properly.

How much credence do biorhythms have? A number of countries apparently swear by them, especially Japan. They

use them in industry as a way of organising their workforce.

Biorhythms have been documented in several television programs, both scientific and discussion. I first heard of biorhythms several years ago on the BBC 1 Tomorrow's World and it was interesting but I did not take much notice of it at the time.

If you would like to get to grips with biorhythms, Imaqlate Software has a program on the market in which you may be interested. It is called *BIO-graph* and produces a graph of your biorhythms for different times of the year — every month, in fact, with a day-to-day analysis of the graph. It takes the work for charting and analysing biorhythms off your hands.

The version for review was on a 3.5in disc and is also available on Microdrive cartridge. It was accompanied by a slender documentation booklet and a paperback book of 128 pages which deals more widely with biorhythms than I could. In the book there are documented graphs of such people as the ill-fated and tragic Janis Joplin, Marilyn Monroe and Judy Garland; all three died from drugs overdoses. There are also graphs of John F. Kennedy, Mario Lanza and Bing Crosby. The book, by the way, is called *Biorhythms*. It was written by Peter West and published by Thorsons Publishing Group. It is being given away with *BIO-graph*, to at least the first 500 customers, or while stocks last.

Accurate

With *BIOgraph* I made graphs of some of the people in the book and checked them against their published graphs — and they matched.

The graphs produced by *BIO-graph* appear to be drawn accurately and accuracy is essential for proper analysis.

There is little need to consult the documentation as *BIO-graph* is well prompted, which makes it easy to operate and

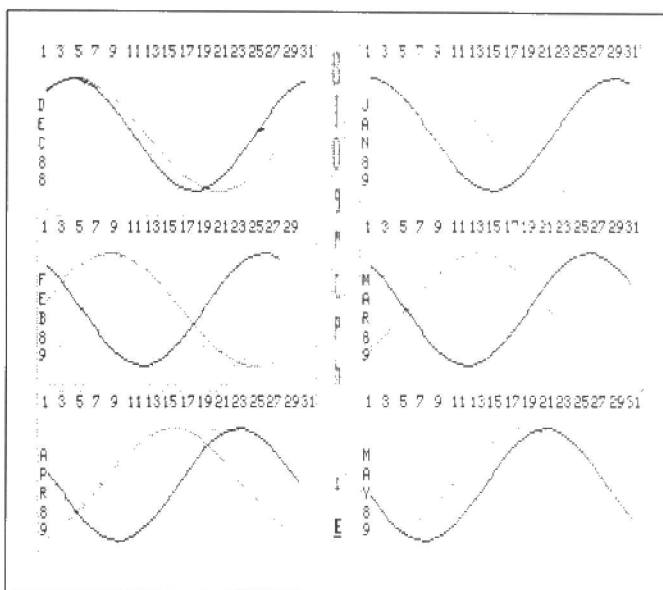
sented by vertical numbered lines. The division between active and inactive is represented by a single horizontal line.

The three cycles are drawn in different colours. Physical is drawn in red, intellectual in green and emotional in white. If you have a monochrome monitor or television set you will be able to distinguish the different cycles easily because of the shades the different colours produce.

You can then select the options by pressing ESC. A menu is superimposed over the graph and the options chosen with a highlight bar which is moved with the up and down cursor keys, with the spacebar for finalising the selection.

Miniature

You can move a month forward or a month back or see a set of miniature graphs for the next six months. There is an option to obtain a day-to-day



the format dd/mm/yyyy, then press ENTER. The final prompt is for when you want the first chart to start. Use the format dd/mm/yyyy and press ENTER.

That is all there is to it. From then it is all menu-driven with prompts in appropriate places to guide you and it is difficult if not impossible to input the incorrect information other than what the program requires.

The biograph of your choice will be drawn on the screen. The graph is for a whole month, with the days repre-

analysis of the chart and to check your compatibility with other people. There are options to print-out the graphs. The screen dumps use GPRINT_prt from the Psion Easel. The author points out that it should be possible to use any of the dump programs from the Easel cartridge so long as they are copied, using the name GPRINT_prt, to your working copy of *BIO-graph*.

All the screens which display graphs can be dumped to printer and hard copies can be obtained of the text which

makes up a chart analysis or a compatibility profile.

When using the compatibility option you will have to enter the names and birth dates of the people you wish to test as being compatible. After the necessary information has been supplied, two graphs will be drawn, one for each person. A compatibility profile can be displayed on-screen or put to printer by selecting the appropriate option from the menu after pressing ESC or any of the cursor keys.

I am assured by the documentation that the compatibility option is great fun at parties. I should think that would be so, because everyone seems to love things which make predictions, I have noticed that with people who are supposed to be able to read tealeaves or palms; for some it is curiosity and for others it is a good laugh.

Biorhythms are not a form of fortune-telling, as they do not attempt to predict what will happen; instead they predict the possible physical, intellectual and emotional states of a person at a given time. Armed with such information it is possible for one to plan ahead to get the best result on the best day, depending to what you wish to apply yourself.

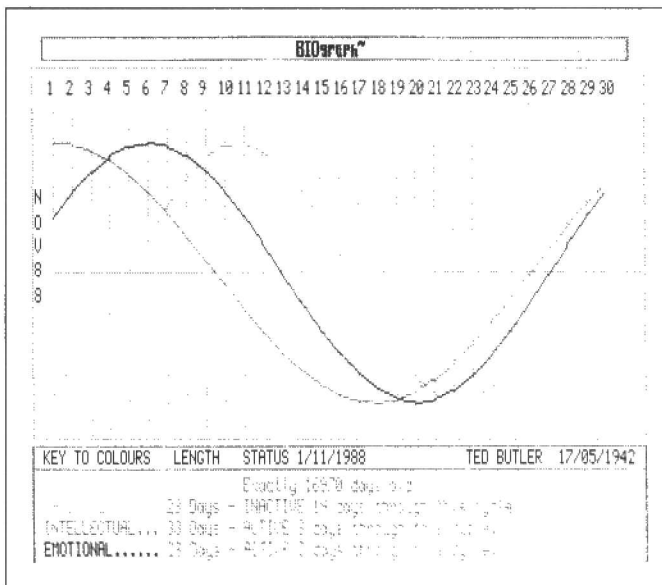
If biorhythms are true then it should be possible for an employer to study the biorhythm

charts of the workforce and put them into compatible groups to do certain types of jobs.

Perhaps a worthwhile update, which could possibly attract interest from employers, would be automatic sorting in the program. I think you would agree it would be a long job to sort through the

compatibility for grouping workers to jobs on certain days.

I used it with the birthdates of people I knew well. I also made biorhythm charts of myself and looked at recent days and events a few years ago I remember well. I felt in certain ways the chart analysis was surprisingly accurate. I also



biorhythm charts, individually, of a large workforce. It would take a good deal of programming to do it but I think it would be helpful if an option were available to load all the available employee details from disc and make an analysis, followed by a printout of com-

patibility of myself with a friend; the result of that was also correct.

I also made a biorhythm chart of someone who became ill that morning and the chart predicted the likelihood of illness being indicated for today for that person.

I have included some sample printouts for a friend I have known for a considerable time and made a chart. They are mainly so you can see what the charts BIOgraph produces are like. The single chart is for November and the six small charts are for the next six months. Also included is an example printout of an analysis produced by the program.

The biorhythms are drawn as sinewaves and through the centre of the graph a line is drawn which is the point where active crosses to inactive, or vice versa, and is also termed as critical. Above that line it is positive and below it is negative. The point just above or below the line is termed semi-critical. Above or below the semi-critical point, depending whether it is positive or negative, denotes whether it is active or inactive.

The bottom of a negative curve of the sinewave denotes that a rhythm is at its lowest

point and the high part of a positive curve is where a rhythm is at its highest point. The worst point is the critical point.

Look at the November chart and follow the physical trace. The fourth shows semi-critical, which indicates a likely lack of energy. The fifth shows critical, a possible black spot for accidents. The sixth shows semi-critical again but as it is in the positive it indicates a possible surge of energy.

From the seventh, it goes active to the 11th, when it is at its peak, after which to the 22nd it falls to the negative low and then journeys upward again to the 30th and beyond. All the time it is going through the highs, active, semi-critical, critical and lows, and so it is for all the rhythms to denote the daily changes which occur to us.

It is not my intention, however, to attempt to make an analysis of the graph but to provide an introduction to the program. BIOgraph is a Turbo-compiled program. It is slick, neat, fast, easy to use and well laid out. Because the program is well-prompted the slender accompanying documentation is adequate and informative to run the program.

I admit when I received BIOgraph I was somewhat sceptical about whether there was anything to biorhythms, or whether it was claptrap. I think I am believing there is something in it.

Entertainment

BIOgraph has a certain amount of entertainment value, I enjoyed using the program and I think I shall, every so often, in the future be making a chart and analysing the possible outcome for days when certain events are looming, even if it is for curiosity and just to see if the nett result tallies with the analysis. Whether it is the kind of program which would entice me to buy it is another matter.

I have a line to draw. If someone writes a program to read tealeaves, the bumps on people's heads or to make clairvoyance easy, and swears it will change my life dramatically, I shall view it with great suspicion.

Ted Butler 17/05/1952 Chart Analysis Date 5/11/1988

PHYSICALLY — CRITICAL

Your potential for accidents is high today. You may also lack mind/body co-ordination. Remember, if driving, take extra care.

INTELLECTUALLY — ACTIVE

Make the most of this positive phase by speaking your mind clearly and effectively. Avoid the possibility of sounding arrogant.

EMOTIONALLY — ACTIVE

As your confidence increases you'll probably find yourself becoming much more talkative. This could be the day to ask for a rise.

Ted Butler 17/05/1942 Chart Analysis Date 6/11/1988

PHYSICALLY — SEMI-CRITICAL

Some people experience surges of energy at this point occasionally followed by extreme fatigue. Pace yourself carefully.

INTELLECTUALLY — ACTIVE

This is a good time to bring to fruition any plans and schemes that you may have postponed. Your ideas are clear and well organised.

EMOTIONALLY — ACTIVE

Today you are at the peak of your cycle. Your most positive aspects are self-confidence and creativity. An excellent day for new ventures and romance.

SOFTWARE FILE

Information:

Program: Home Banker and Home Banker+
Supplier: DJW Software,
11 Pound Close, Bramley,
Hampshire RG26 5BL.
Tel: 0256 881701.
Price: £14.95 and £15.95

We all hate bank charges but what can we do about them? Part of the answer lies in the new *Home Banker* package from DJW Software. Banks have, of course, been using computers to keep track of your money for many years, which is why they have had the upper hand, delaying the clearing of cheques paid to you by several days but charging you instantly for any withdrawals. It is amazing how careless people are about their bank accounts, even if for a job or business they are meticulous with every penny.

The reason most people do not bother about their current account, except to work themselves into a rage when

the statement arrives, is because it is too difficult to keep track of all those direct debits, standing orders, cash dispenser withdrawals and cheques which may or may not have been cashed. If you have a joint account with two cash cards the problem of keeping in credit is increased twentyfold.

Enter the QL and Home Banker software to fight new technology with new technology. It is supplied in two versions, one on Microdrive cartridge for the basic 128K QL owner and the Plus version on disc for those with disc drives and extra memory. A total memory in excess of 256K is needed for satisfactory operation of Home Banker+.

The different versions are a reflection of the problem which

Andrew Shephard looks at a money program which controls as well as counts.

faces all software writers for the QL — how to produce a package which is suitable for a wide range of specifications in terms of memory and storage devices. The result is often a

compromise between what can be run on the basic QL and what could be run on a machine with the maximum memory expansion. By offering two versions, the compromises are kept to a minimum.

Home Banker, in effect, creates a parallel record of the accounts you might have with banks or building societies. It allows you to have instant information about the state of your balance, a statement — on-screen or printed-out — covering any interval you choose and analysis of different areas of expenditure. The first two can be provided by a visit to the bank, cash dispenser, or in some cases by telephone access to an account, but the analysis is an additional service not available from banks.

Easy

This is an easy package to get to know. A small user guide supplied with early versions has been upgraded to provide greater readability, though this is not a package which requires

much user experiment. Error messages and sound warnings are used to good effect to keep the account holder on the proper track.

After loading, the user is asked for a PIN number and without it the program will not proceed. This is a security measure to prevent unauthorised access. The PIN number is supplied at the back of user guide but presents me with two problems. How can I remember yet another PIN number when I keep forgetting the ones I already have and who would be interested in my bank account? Still, the PIN adds a touch of excitement to the mundane business of bank accounts.

Having opened the security code, the user is presented with 12 or 19 options, depending on which version is loaded. They are selected by cursor to highlight a particular option. I would have liked a single-letter keypress as an alternative to moving round the screen by cursor keys. Opening an account is necessary before any details can be entered and that is done by giving it a name. Transactions are then entered as receipts or payments, with each transaction requiring a

home
BANKER

Please Type in Your PIN Number

home
BANKER

Use + + + to choose transaction and ENTER to select.
Press F1 for Help or Press "Q" to leave Home Banker

Payment	Reconcile	Open Account	Save File
Receipt	Delete Record	Close Account	Load File
Account Balance	Change Record	Search	Colours
Statement	Analysis Balance	S/Os	Printer

Number of Transactions remaining: 2712/2712

source, a description, a category, an amount and a date. You are also asked if the transaction is to be reconciled or not.

That is where the package has the advantage over less-sophisticated bank account programs. The matter of reconciliation is important to the accuracy of the current balance. Cheques which are not yet cleared, whether going in or out of the account, should be entered as unreconciled. The 'reconcile' option allows you to go through all the previously-uncleared cheques at a later date, reconciling those which should now be cleared.

Guesswork

In practice, there is an element of guesswork in how long it takes a cheque to clear, or how long it takes some firms to cash the cheques you have sent them. The 'balance'

file is loaded the standing orders are 'generated', which means that all those which fall within the period at which you are looking will have appeared on your statement.

This option is clever and also essential to the practical value of Home Banker as a tool for keeping out of the red. Version 1 does not have this option, which limits seriously the value of Home Banker on the basic QL. Of course, it is still possible to enter each occurrence of a standing order as a separate transaction but it is easy to forget those nasty quart or yearly ones. Other features available only on Home Banker + are an option to calculate interest on savings accounts and wider range of printed statements.

Analysis of home expenditure is made possible by careful use of the categories under which payments are entered. This is one area where the user guide could have been more forthcoming, especially as you

create a different category and spoil your analysis.

Accounts

The package can cope with up to 10 accounts, which should satisfy those who do not like putting all their eggs into one basket. All accounts are saved under one filename and are saved and loaded together. They are numbered and the user has to select the relevant number before each transaction.

I tried Home Banker on three systems. On a basic QL, using Microdrives as my temporary storage medium, the program takes about half a minute to load and allows slightly more than 400 transactions, which might be enough for a year's banking. The Microdrives are in constant use but this does not delay the operation of the program. Even using a colour TV and F2 mode, the screen displays are clean and clear.

On an expanded QL 512K Memodisc expansion — with disc drives — an extra line has to be added to the boot program to create a RAMdisc as the temporary storage device. "Format RamL_8" allows 256K of temporary storage, allowing more than 1,000 transactions.

People who write many cheques or want to analyse expenditure over long periods may want to make use of the

2,720 transactions possible with a Trump Card expansion — 768K. Trump Card allows a RAMdisc to be created by accessing it, so there is no need to modify the boot program of Home Banker in this case. DJW Software offers back-up to anyone who experiences difficulties on a particular system.

It would probably take most people only a few minutes a week to enter all account transactions. Real computer banking freaks have probably already opened an account which can be accessed directly by computer, which some banks offer. At present, such a service costs extra. Home Banker should be a saver.

Doubters

If any software will convince doubters that it is worth keeping tabs on their personal finances, Home Banker will do it. It is a pleasant program to use, needs no specialist knowledge of banking or accounts, and the routines for entering transactions are fast, once familiarity is achieved.

Perhaps you do not want to know how much you spend on computer hardware and software but if Home Banker helps you keep in the black, even for three months of the year, you will certainly save the cost of this package in one position of nil bank charges.

home BANKER

Change & record of R J & A J Shepard

Reference number: 3
1. Date: 26 10/1988
2. To: Self
3. Description: Cash
4. Analysis Category: Food etc
5. Amount: £50.00 DB
6. Reconciled: Yes
7. Transaction No.: 589.1

Press 1,2,3,4,5,6 or 7 to alter or ESC to end

home BANKER

Use ←→ to choose transaction and ENTER to select.
Press F1 for Help or Press 'Q' to leave Home Banker

Payment	Reconcile	Open Account	Save File
Receipt	Delete Record	Close Account	Load File
Account Balance Change Record	Search	Colours	
Statement	Analysis Balance	S/Os	Printer

Number of Transactions remaining: 2678/2712

option gives two figures, a pessimistic one assuming that all transactions have taken place and an optimistic one telling you how much of the balance should be available.

Home Banker + allows you to enter standing orders, giving the frequency of the payments and the day of the month they are paid. A quick look at your previous bank statements will give you this information. The standing orders and other transactions are saved under a filename and the next time the

cannot see the categories on-screen you have chosen. The guide should warn you to work out your main areas of expenditure before starting to enter transactions.

Analysis

As with any kind of analysis, you need to work out what you want to know before you start. This is a pencil and paper job; be careful to enter categories in exactly the same way. A full-stop at the end of a word will

A Better Search

I Roger Strickland demonstrates that the obvious way is not necessarily the best way for search algorithms.

This article questions the assumption that the algorithm to search for text strings — as in word processors) is intuitively obvious — and demonstrates a more sophisticated and efficient method, using the Boyer-Moore algorithm. A full demonstration program, coded for the QL, is presented for the reader to experiment.

Could there be a simpler operation than searching a piece of text for a simpler operation than searching a piece of text for a specific character? The concept is so trivial that it is the kind of task given to a programming beginner as an exercise.

What about searching for a specific string of characters in a piece of text? Intuitively that is just as simple in concept, although obviously a little more code needs to be written. It is in this slightly more complicated case that intuition lets us down. The 'obvious' way of coding such a search is the incorrect approach. It will not provide us with the most efficient method.

The first time the new search technique is encountered it usually causes wonderment. When I first met it I had been a professional programmer for several years and I think I found the event chastening. If I could believe that no-one could teach me about something so fundamental as a string search, what other mistaken beliefs did I hold?

As an example is the best way of illustrating the method, let us take as our text string the 26 characters of the alphabet, in normal sequence. If we wish to search for a specific character, say 'x', we start by comparing that character to the first character of the next string, 'a'. If the match fails, we advance by one position along the text string and try again. The search terminates when either a match occurs or we have reached the end of the text string.

Now let us take a more general case, where we are searching for more than one character, say 'vwx'. We shall call these characters the search string. A formal description of the (traditional) search method is:

- 1) Align the start of the search string with the start of the text string:

```
abcdefghijklmnopqrstuvwxyz  
vwx
```

- 2) Scanning from the start — left-most end — of the search string, compare each character in turn with the corresponding character in the text string. If a pair of characters matches, repeat the

comparison with the next — one position to the right — pair. Terminate the comparison when either (a) the last — right-most — character of the search string has been matched successfully; treat this as a success, or (b) a pair of characters do not match; treat this as a failure; or (c) a character in the search string is unable to pair with a text string character because the text string is too short; treat this as a terminating failure.

- 3) If operation 2 has terminated successfully, the search is over with the search string being located in the text string. Conversely, if the operation has ended with a terminating failure, the search is again over, with the search string not found in the text string.

In the case of the final option, where there is a simple match failure the search string should be advanced by one character position to the right in relation to the text string and operation 2 should be repeated:

```
abcdefghijklmnopqrstuvwxyz  
vwx
```

In this instance the search string will be advanced 21 times along the text string and there will be $21 \times 1 + 1 \times 3$ comparisons before the search terminates successfully. Are all of these 24 comparisons really necessary?

It would seem that the original example described in the previous section, searching for the single character 'x' in the text string, could not be improved. There are 24 comparisons. Apart from some additional knowledge about the text string at the start, like the fact that the character being searched for occurs in the right half of the string, and therefore it would be more profitable to search from the right-hand end, there seems nothing to be done which could improve the search time consistently. Even given the prior knowledge that the character 'x' appears in the right half of the string is not useful unless we know it occurs only once in the whole text string; after all we are trying to find the first occurrence of the character from the left-hand end.

We can reach a different conclusion about the second example when we are searching for a match with a search string longer than one character. Look at the first three alignments made during the classical search:

```
abcdefghijklmnopqrstuvwxyz  
vwx
```

```
abcdefghijklmnopqrstuvwxyz  
vwx
```

```
abcdefghijklmnopqrstuvwxyz  
vwx
```

For each alignment, the character 'c' in the text string is paired with one of the search string characters. Now 'c' is not contained in the search string, so after the first alignment we need not even bother to carry-out the next two — they must fail. We could advance the search string by three character positions after the first failure. The same would happen with the next comparison, when the search string could be advanced by another three character positions.

Now it should be obvious that this can work only if we compare the last — right-most — character in the search string first instead of the left-most. Thus the search string is compared in a character order of 'x', 'w', 'v'.

Further, we must have a simple method of knowing that the character 'c' does not occur in the search string. This is accomplished by means of a look-up table. This table, of size 256 entries — one for each possible 8-bit character — initially is filled with zero values. Then the search string is scanned and for each letter which occurs in it, the corresponding character position in the table is filled with a non-zero value. The search proceeds as follows:

```
abcdefghijklmnopqrstuvwxyz  
vwx
```

Scanning from the right, 'x' in the search string is compared to 'c' in the text string. The match fails. The table entry for 'c' is located; it is zero-valued, so the search string is advanced by its complete length along the text string, and we arrive at the next position:

```
abcdefghijklmnopqrstuvwxyz  
vwx
```

The same process occurs as before. A total of seven such shifts occur before we have the desired alignment and three more comparisons conclude the match, a total of 10 comparisons versus the 24 of the previous method.

Of course, although we have reduced the number of comparisons, we have now introduced two overheads. First, the setting-up of the look-up table. As this is a one-off operation, independent of the length of the text string, the cost of this may be ignored if the text strings are sufficiently long.

Second, we have introduced a table look-up with each failed comparison. The cost of the look-up is comparable with the cost of a comparison. So, broadly, with the example given we have approximately the equivalent of table set-up time plus 17 compares. It may, in fact, be slower than the usual string comparison method.

Continued on page 30

The real savings occur, however, when the search string becomes longer. We saw that the search string was shifted by its own length each time a comparison of the final character failed. If we had been searching instead for the string 'nopqrstuvwxyz' there would have been a total of one shift — of 13 places — and 14 comparisons, compared to the classical case of 13 shifts and 26 comparisons.

Now having the idea of the technique, let us turn it into a proper working model. We have to consider not just the last character position in the search string but every position. Also what should happen if the current character in the text string occurs in the search string but not at the current character position of the search string?

We shall build the search algorithm in stages, using more complex examples at each stage and showing the SuperBasic code necessary to have a true working system. The method uses the Boyer-Moore algorithm (1976).

For our search string we shall use the characters 'abcdef' and for the text string we will show each character currently not of interest to use as a '.' and each relevant character as it really is. Thus we may have the text and search strings aligned as:

```

. . . . . a . . . . .
a b c d e f

```

It is obvious that following the failure to match the 'a' and 'f', the search string should be advanced by five places to the right. If the text string character had been a 'b', the advance should be four places, a 'c' three places and so on.

The rule appears to be that the amount of advance when a match fails but the current text character exists in the search string, is by the number of character positions between the current non-matching search string character ('f') and the search string character which matches ('a'). If we number the character positions in the search string:

```

. . . . . a b c d e f . . .
1 2 3 4 5 6

```

the distance to move is the difference between the numbers of the two character positions, i.e., $6-1=5$.

This is the shortest stage. Using the example again, how much must the search string be shifted if the character in the text string does not occur in the search string?

```

. . . . . m . . . . .
a b c d e f

```

Obviously, six places. This is easier to consider as the difference between the number of the current character position in the search string (6) and the number of the character position before the start of

the search string (0).

```

. . . . . m . . . . .
a b c d e f
0 1 2 3 4 5 6

```

This non-existent character may be considered to match any text string character not already included in the search string. It makes the algorithm easier to picture thus.

So far we have looked only at the last character position of the search string. Let us suppose that it matches the corresponding character in the text string. We therefore move one position to the left and compare the next pair of characters:

```

. . . . . b . . . . .
a b c d e f
. . . . 0 1 2 3 4 5 6

```

Notice that the shift distance required is the difference between the numerical position of the 'e' and the 'b' in the search string, i.e., $5-2=3$. This is the same calculation as for the right-most character position in the search string. The technique is completely general; it works for all the character positions in the search string.

The search program will always know what is the number of the current character in the search string. How does it know the number of the other character position in the string? By using the look-up table, of course.

Therefore we will refine the settings in the look-up table. All entries representing characters which do not exist in the search string will be set to numeric position 0 — exactly as we proposed setting them at the beginning. The other characters will be set to their actual position in the search string, treating the left-most character position as 1. Setting-up the table in SuperBasic is thus very straightforward:

```

DIM table(256)
FOR a=1 TO 256:table(a)=0

```

By default the table will be filled with zero valued entries anyway following a DIM statement, so the 'for' loop is not needed. If 'table' were to be re-used for a later search it would need to be re-initialised using this or a similar method.

```

FOR a=1 TO LEN(search$)
table(CODE(search$(a TO a))) = a
END FOR a

```

Let us now write the short piece of code which carries-out the search. We shall let the variable 't' represent the position in the text string against which the left-most character of the search string is aligned. We shall let the variable 's' represent a position within the search string. e.g.:

```

. . . . .
a b c d

```

t has the value 5 in this case.

```
t=1
```

start at the left

REPEAT loop

once round the loop for each alignment of the search string.

```
IF t+LEN(search$)-1 > LEN(text$):E
XIT loop
```

if end of search string off the end of the text, give up.

```
FOR s=LEN(search$) TO 1 STEP -1
```

repeat inner loop for each char in the search string — from the right end.

```
IF text$(t+s-1) <> search$(s TO s)
```

... advance the search string using the look-up table.

NEXT loop

and exit from the inner loop.

```
END IF
END FOR s
```

otherwise pair of chars match, so let us try the previous pair.

```
PRINT "match found at position
":t:STOP
```

can get here only if a successful match of all chars in search string.

```
END REPEAT loop
PRINT "no match found":STOP
```

Ah, well

There is only one more point to make. What happens if the same character occurs more than once in the search string? What value should be put in the look-up table? e.g.:

```

. . . . . b . . . . .
a b c b d

```

In viewing the problem, it is obvious that the search string should be advanced by one position align with right-most 'b', rather than advancing three positions to line up the left-most.

We therefore require that the look-up table should contain the number of the right-most occurrences of each character in the search string. As we set up the look-up table by inspecting the search string one character at a time from the left-hand end, later occurrences of a character which has already been placed in the look-up table will over-write the value with the new one. So all is satisfactory after all. Well, almost. We have neglected the following possibility:

```

1000 REMark *****
1010 REMark * Program to demonstrate visually how an efficient string *
1020 REMark * search algorithm works. The algorithm is based on the *
1030 REMark * Boyer-Moore algorithm. Roger D. Strickland Nov 88 *
1040 REMark *****
1050 :
1060 REMark *****
1070 REMark * Main control program. After initialisation, a repeat *
1080 REMark * loop is entered, each iteration of which gives a *
1090 REMark * complete example of searching multiple times for a user *
1100 REMark * specified search string. Each example itself contains *
1110 REMark * a repeat loop, each iteration of which causes a search *
1120 REMark * to be made for the NEXT occurrence of the search string *
1130 REMark * in the text. *
1140 REMark *****
1150 initialisel
1160 REPEAT example_loop
1170     initialise2
1180     setup_lookup_table(search$)
1190     REPEAT next_match_loop
1200         pos=search(search$,text$,pos)
1210         IF pos = 0
1220             AT 18,0:CLS 3:AT 18,0:PRINT "end of search"
1230             EXIT next_match_loop
1240         ELSE
1250             AT 6,pos-1:PRINT search$
1260             AT 8,pos-1:PRINT "-";
1270         END IF
1280         pos=pos+1
1290         increment_shift_count
1300     END REPEAT next_match_loop
1310     AT 19,0:PRINT "Press ENTER for another turn or ESC to exit"
1320     REPEAT key_loop
1330         k=CODE(INKEY$(-1))
1340         IF k = 10:EXIT key_loop
1350         IF k = 27:STOP
1360     END REPEAT key_loop
1370 END REPEAT example_loop
1380 :
1390 REMark *****
1400 REMark * initialisel. Procedure that carries out once-only *
1410 REMark * initialisation - ie at start of program run. *
1420 REMark * This includes dimensioning the variables, setting up *
1430 REMark * the (fixed) text in text$, displaying the title and the *
1440 REMark * 'ruler' above where the fixed text will be displayed. *
1450 REMark *****
1460 DEFINE PROCEDURE initialisel
1470     DIM table(256), search$(100), text$(100)
1480     text$="skinning bananas can be skilled but look out as you can be
        killed"
1490     MODE 4:CLS:INK 4:CSIZE 2,1:AT 0,13:PRINT "FAST SEARCH"
1500     CSIZE 0,0:INK 4:AT 4,0:FOR a=1 TO 7:PRINT "          ";a;
1510     AT 5,0:FOR a=0 TO 6:FOR b=1 TO 9:PRINT b;:NEXT b:PRINT "0";:NEXT a
1520 END DEFINE initialisel
1530 :
1540 REMark *****
1550 REMark * initialise2. Procedure that carries out required *
1560 REMark * initialisation once per example. *
1570 REMark * This includes displaying the fixed text string, *
1580 REMark * prompting for and getting the user defined search *
1590 REMark * string into search$, clearing down the shift and *
1600 REMark * compare counts to 0 and displaying them, and waiting *
1610 REMark * for the user to hit a key. *
1620 REMark *****
1630 DEFINE PROCEDURE initialise2

```

Continued on page 32

Continued from page 31

```
1640 AT 6,0:INK 0:PRINT text$
1650 AT 6,0:CLS 2
1660 AT 10,0:INK 7:PRINT "enter search string: ";:INPUT search$
1670 AT 10,0:CLS 3
1680 compare_count=0
1690 shift_count=0
1700 AT 12,0:PRINT "shifts compares"
1710 AT 14,3:PRINT "0 0"
1720 AT 19,0:PRINT "Press any key for each move"
1730 pos=1
1740 END DEFine initialise2
1750 :
1760 REMark *****
1770 REMark * draw_search_string. Procedure that prints the user *
1780 REMark * defined search string in the current position beneath *
1790 REMark * the text, and waits for the user to hit a key. *
1800 REMark *****
1810 DEFine PROCedure draw_search_string(p)
1820 AT 7,0:CLS 3
1830 AT 7,p-1:PRINT search$
1840 k=CODE(INKEY$(-1))
1850 END DEFine draw_search_string
1860 :
1870 REMark *****
1880 REMark * increment_shift_count. Procedure that increments the *
1890 REMark * count of the number of shift operations so far and *
1900 REMark * displays the current value. *
1910 REMark *****
1920 DEFine PROCedure increment_shift_count
1930 shift_count=shift_count+1
1940 AT 14,3:PRINT shift_count
1950 END DEFine increment_shift_count
1960 :
1970 REMark *****
1980 REMark * increment_compare_count. Procedure that increments the *
1990 REMark * count of the number of compare operations so far and *
2000 REMark * displays the current value. *
2010 REMark *****
2020 DEFine PROCedure increment_compare_count
2030 compare_count=compare_count+1
2040 AT 14,10:PRINT compare_count
2050 END DEFine increment_compare_count
2060 :
2070 REMark *****
2080 REMark * setup_lookup_table. Procedure that sets up a 256 entry *
2090 REMark * table specific to the search string chosen by the user. *
2100 REMark * For details, refer to the accompanying article. *
2110 REMark *****
2120 DEFine PROCedure setup_lookup_table(s_search$)
2130 LOCAL a
2140 FOR a=1 TO 256:table(a)=0
2150 FOR a=1 TO LEN(s_search$)
2160 table(CODE(s_search$(a TO a)))=a
2170 END FOR a
2180 END DEFine setup_lookup_table
2190 :
2200 REMark *****
2210 REMark * search. Procedure that is at the heart of the program. *
2220 REMark * Given as parameters, the user specified string to be *
2230 REMark * searched for, the fixed text which is to be searched, *
2240 REMark * and the start position in the fixed text where the *
2250 REMark * search is to commence, the procedure returns either the *
2260 REMark * position in the fixed text where the search string is *
2270 REMark * found, or zero to indicate that no match was made. *
```

```

2280 REMark * During the search, the print procedures are called so *
2290 REMark * the user may see the progress of the search. As these *
2300 REMark * procedure calls are not part of the search algorithm *
2310 REMark * proper, they are highlighted by a line of asterisks in *
2320 REMark * the source, so that they do not become confused with *
2330 REMark * the functional code. For details of how this procedure *
2340 REMark * works, refer to the accompanying article. *
2350 REMark * To simplify understanding, this routine has NOT been *
2360 REMark * optimized. That should certainly be done if the reader *
2370 REMark * wants to use the algorithm seriously in his own *
2380 REMark * programs. *
2390 REMark *****
2400 DEFine FuNction search(s_search$, s_text$, s_start)
2410   LOCAL s,t,shift
2420   t=s_start
2430   REPEAT align_text_loop
2440     draw_search_string(t):REMark *****
2450     IF t+LEN(s_search$)-1 > LEN(s_text$):RETurn 0
2460     FOR s=LEN(s_search$) TO 1 STEP -1
2470       increment_compare_count:REMark *****
2480       IF s_text$(t+s-1 TO t+s-1) <> s_search$(s TO s) THEN
2490         increment_shift_count:REMark *****
2500         shift=s-table(CODE(s_text$(t+s-1 TO t+s-1)))
2510         IF shift < 0 THEN
2520           t=t+1
2530         ELSE
2540           t=t+shift
2550         END IF
2560       NEXT align_text_loop
2570     END IF
2580   END FOR s
2590   RETurn t
2600 END REPEAT align_text_loop
2610 END DEFine search

```

```

. . . . . b . . . . .
a b c b d

```

There exists in the search string a character which matches the current text string character but is to the right of it. Using the look-up table method to determine the required shift, we get a negative number. That would signify moving the search string to the left, something we definitely do not want to do. We could have the search program oscillating the search string indefinitely about some region of the text string.

Ideally, we would like the look-up table to contain the position of the next occurrence to the left of the character 'b' — position 2 instead of position 4. Although such dynamic fiddling with the look-up table is possible, it is likely that the processing involved exceeds that saved by using this method. As a compromise, the following solution is suggested.

When the look-up table indicates a search string move by a negative amount, move the search string one position to the right. The code presented here is a search routine obeying all of the rules described so far. There are two routines. The first is

a procedure, 'setup_lookup_table', which creates the look-up table. The second is a function, 'search' which, when given the character position in the text string at which to start the search, returns either the value of the character position in the text string where the next match occurs, or 0, indicating that the substring was not found before the end of the text string was reached.

```

DEFine PROCedure setup_lookup
__table(s__search$)
  LOCAL a
  FOR a=1 TO 256:table(a)=0
  FOR a=1 TO LEN(s__search$)
    table(CODE(s__search$(a TO a)))=a
  END FOR a
END DEFine setup_lookup__table
DEFine FuNction search(s__search$,
s__text$, s__start)
  LOCAL s,t,shift
  t=s__start
  REPEAT align__text__loop
    IF t+LEN(s__search$)-1 >
    LEN(s__text$):RETurn 0
    FOR s=LEN(s__search$) TO 1
    STEP -1
      IF s__text$(t+s-1 TO t+s-1) <>
      s__search$(s TO s) THEN
        shift=s-table(CODE(s__text$(
        (t+s-1 TO t+s-1)))

```

```

    IF shift < 0 THEN
      t=t+1
    ELSE
      t=t+shift
    END IF
  NEXT align__text__loop
END IF
END FOR s
RETurn t
END REPEAT align__text__loop
END DEFine search

```

This section contains a fully-functional program which will display visually the shifts and comparisons which take place on any string the user chooses. The program presents a line of text and the user is prompted to input a search string. A search is then made for every occurrence of the search string in the text, with each shift operation and set of compares being under the control of the user.

A running total of the number of shifts and character comparisons is maintained, which demonstrates that the method works well in practice.

Choosing a search string of 'ana' gives an interesting result — it shows that strings searched for may overlap each other in the text but will still be found.

The user may, of course, edit the program to change the fixed text string.

The QL came into being equipped with two small tape recorders called Microdrives which in many instances perform satisfactorily. There have been problems of reliability mainly due to excessive heat, incompatibility between drives, expense of the tiny cartridges, while the more attractive proposition of using disc drives is always worth consideration.

By using disc drives there are advantages of increased storage capacity, much faster access, increased reliability and slowly falling prices. Before a disc drive can be hooked to a QL an interface must be obtained, either with or without extra memory. It will have some software incorporated on a ROM, usually by Tony

DISC DRIVE DIRECTIONS

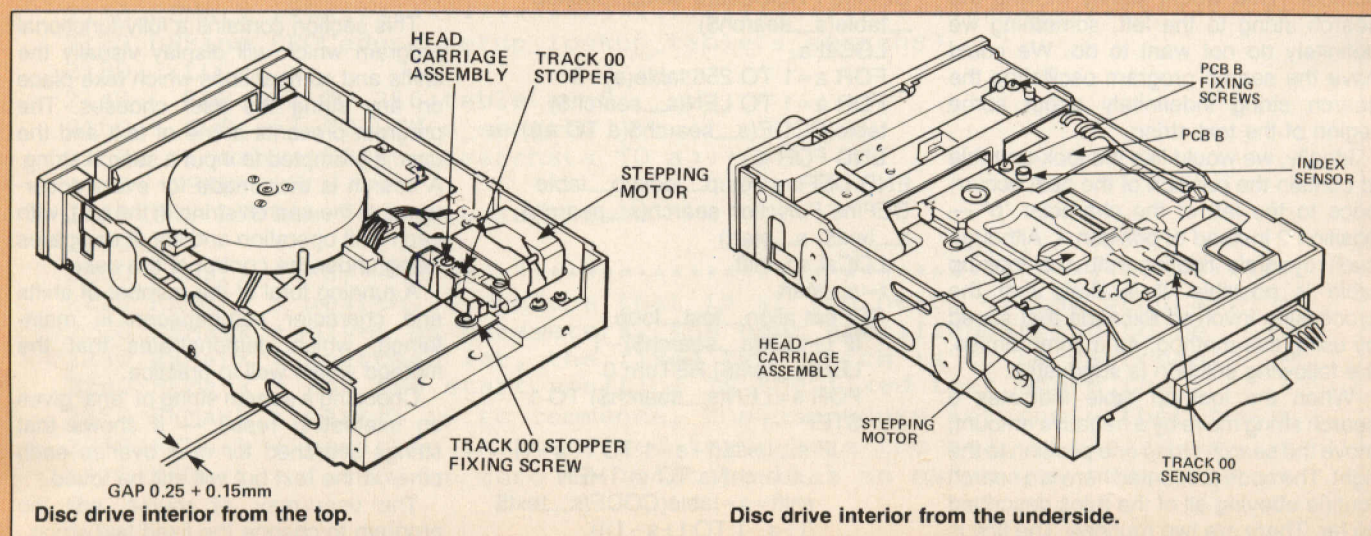
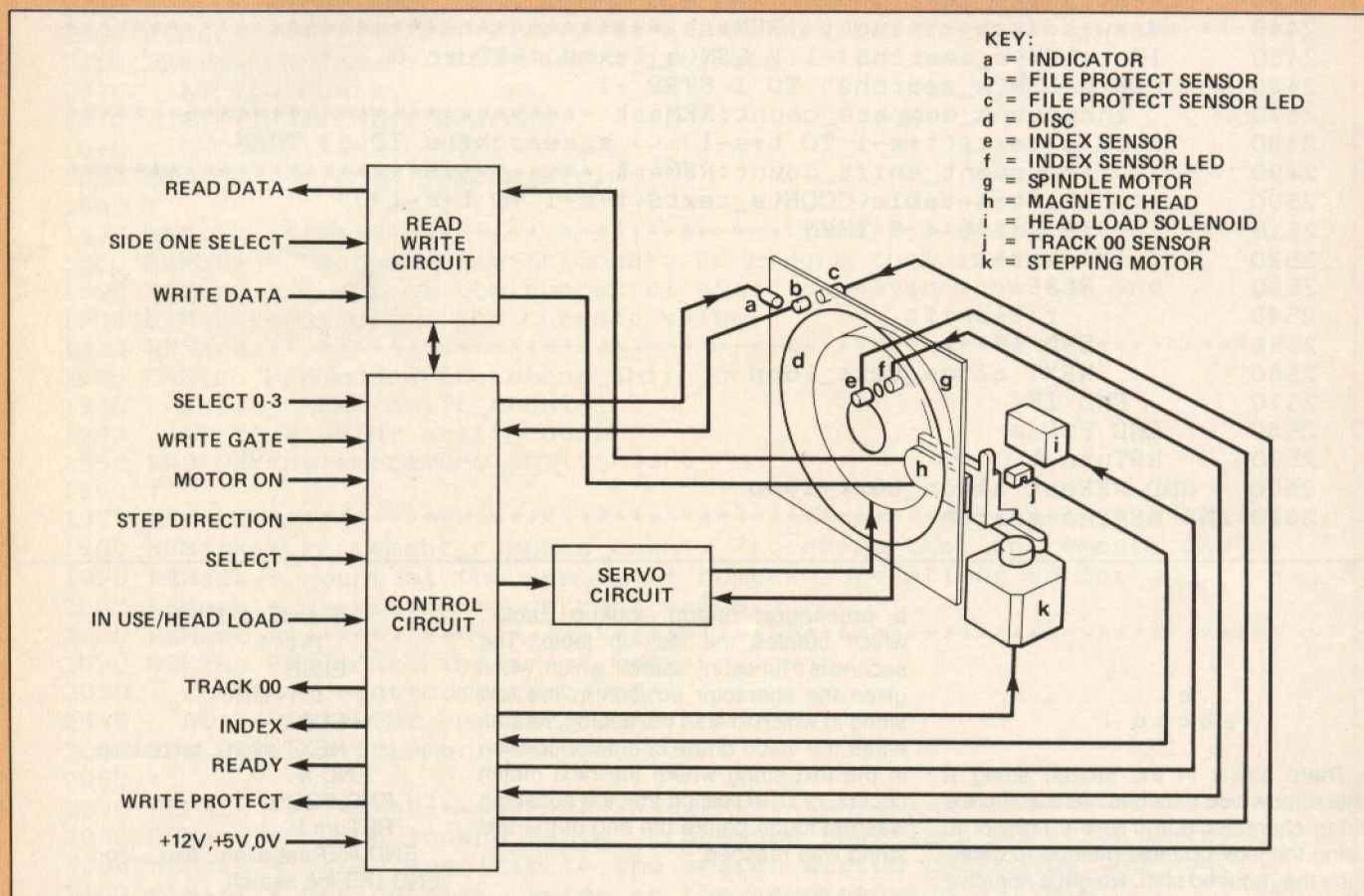
Dennis Briggs presents some basic advice on using disc drives with the QL.

Tebby, which provides all the necessary housekeeping and leads to some form of standardisation.

Usually some or all of Toolkit 2 is incorporated in the ROM. If the ROM is an early version it is advisable to contact QJump to see if there is an updated ROM

available. In effect, the bit out of the end with the ribbon cable plugged in allows you full control over the disc drives. Most interfaces allow up to four drives to be plugged in, with the exception of the Trump card which will address only two.

The drives are daisychained down the



ribbon cable with a resistor block being present in the last drive. Virtually no extra power is needed for this arrangement, as only one drive can be active at any time.

Let us make things clear in regard to what the trade calls rotating media. There are all kinds of physical sizes such as 8in., 5.25in., 3.5in., 3.25in., 3in. and so on which can be used on the QL. The only ones I have encountered which are not suitable are early Apple drives using hard-sectored discs and Commodore drives which are a law unto themselves.

Hard-sectored discs look the same as soft-sectored ones but have more index holes. I have used both an old Atari and Dragon drives with a little work, while BBC drives are just plug-and-go once you have the interface. Eight-inch drives need two wires changing over while 3.25in.-drives need discs which are not available in the U.K., so the question is what drives will work with the QL? The answer is to plug in one and try it. Can you do harm by this approach? No, provided there is no disc in the drive. If there is it may write rubbish to it at that point.

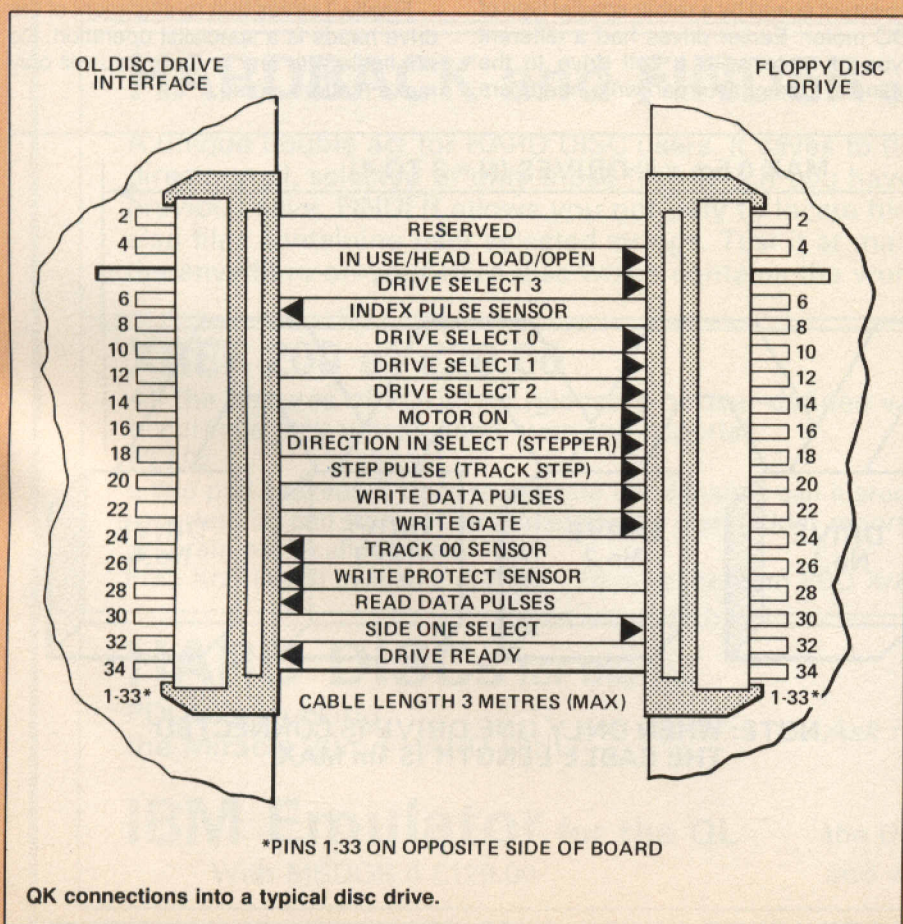
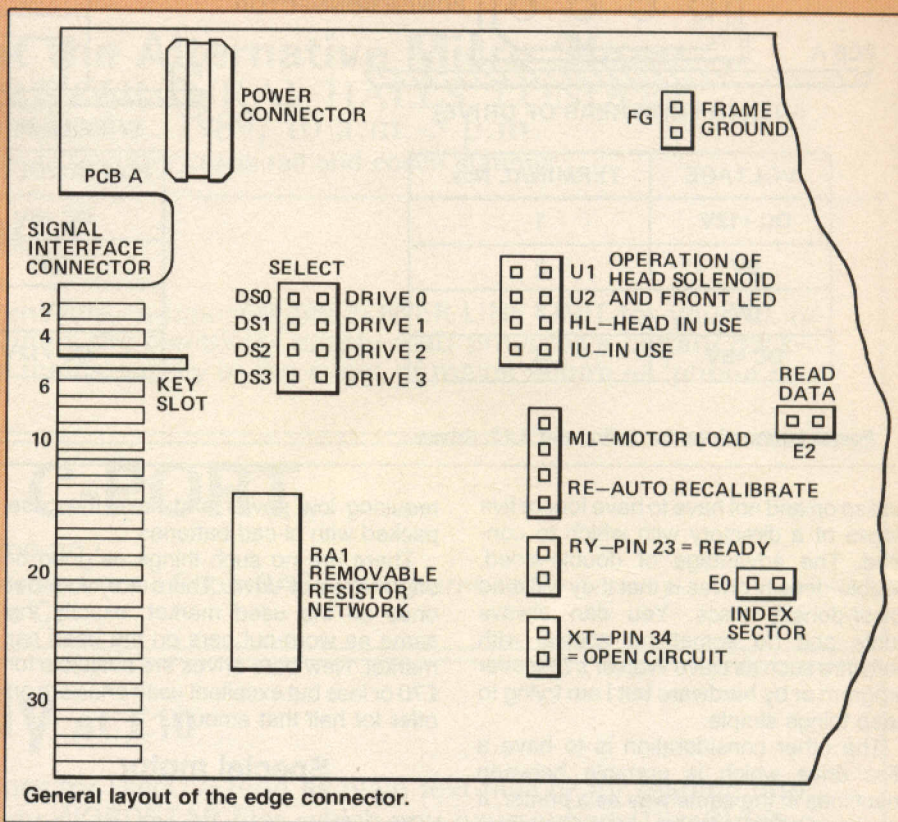
Medic had an unhappy experience when it decided to feed power down the 34-way signal cable and got the connector the wrong way round. Its interface was probably the best one for the QL in relation to what can be done with it. Keep the power and signal lines separate or you will zap the interface chips. CST also managed to get the interface connector upside down on some boards.

The connector between the interface is

a 34-way ribbon cable of any colour — or even rainbow colour. It does not make it work better in colour but it is good to know which is line 1, the line with the stripe on the left-hand side. It goes to pin 1 on the interface and pin 1 on the disc drive. You

could turn it round so that pin 1 on the interface connects to pin 34 on the disc drive but it will not work, neither will it do damage.

Notice that every other line is connected to ground, to screen one signal



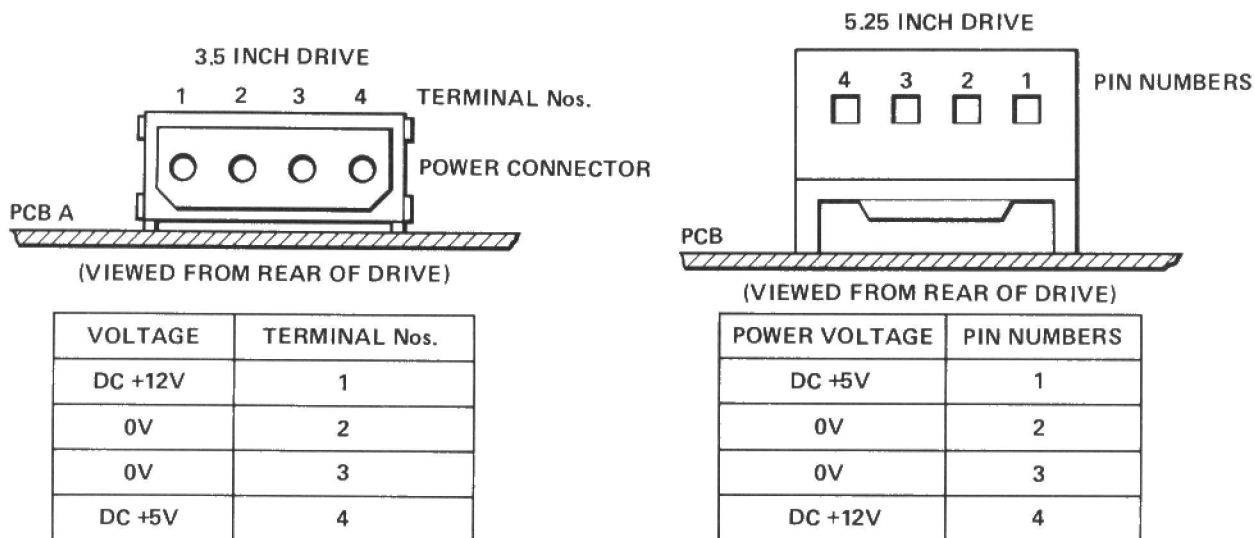
line from its neighbour, so if the connector is fitted upside down all the signal lines are connected to ground.

The disc drive of whatever make or size will require power for the motor and power for the electronics. It needs a regulated 12V and 5V unless you have one of the latest expensive Micropower drives which will manage on 5V only. The power cannot be derived from the QL.

Assembly

Many companies offer a ready-packaged disc drive ready to plug in and go, all giving good performance at a price. At a much lower price it is possible to buy two smaller capacity drives with a case and a power supply but it will mean you have to do some simple assembly work. A single-sided, single-density drive will format to 360 sectors, which is increased to 720 sectors by a single-sided, double-density drive with a further doubling to 1,440 sectors by a double-sided, double-density drive. A further increase is not possible on the QL and the drives on other machines are not particularly reliable.

There are valid reasons for opting for lower capacity drives and also for considering 5.25in. units. The main one is that of lower cost, both of the drive and the disc, with the ability to read discs from foreign machines such as BBC, PC clone



Power connections for 3.5in. and 5.25. drives.

and so on and not have to have four or five pages of a directory with which to contend. The advantage of double-sided, double-density drives is that they will read lower-density discs. You can always fiddle and do something special with software such as Dave Walker's *Discover* program or by hardware but I am trying to keep things simple.

The other consideration is to have a disc drive which is portable between machines in the same way as a printer. If you have a BBC, Amiga, Atari, Spectrum or anything reasonable it can be done easily. While mentioning portability, it is possible to obtain Epsom 3.5in. drives which work from a battery. They are ones

requiring low power and have the case packed with ni-cad batteries.

There are no such things as good or bad makes of drive. There are worn-out ones on the used market, exactly the same as worn-out cars on the used car market. New bare drives are available for £70 or less but excellent used ones are on offer for half that amount.

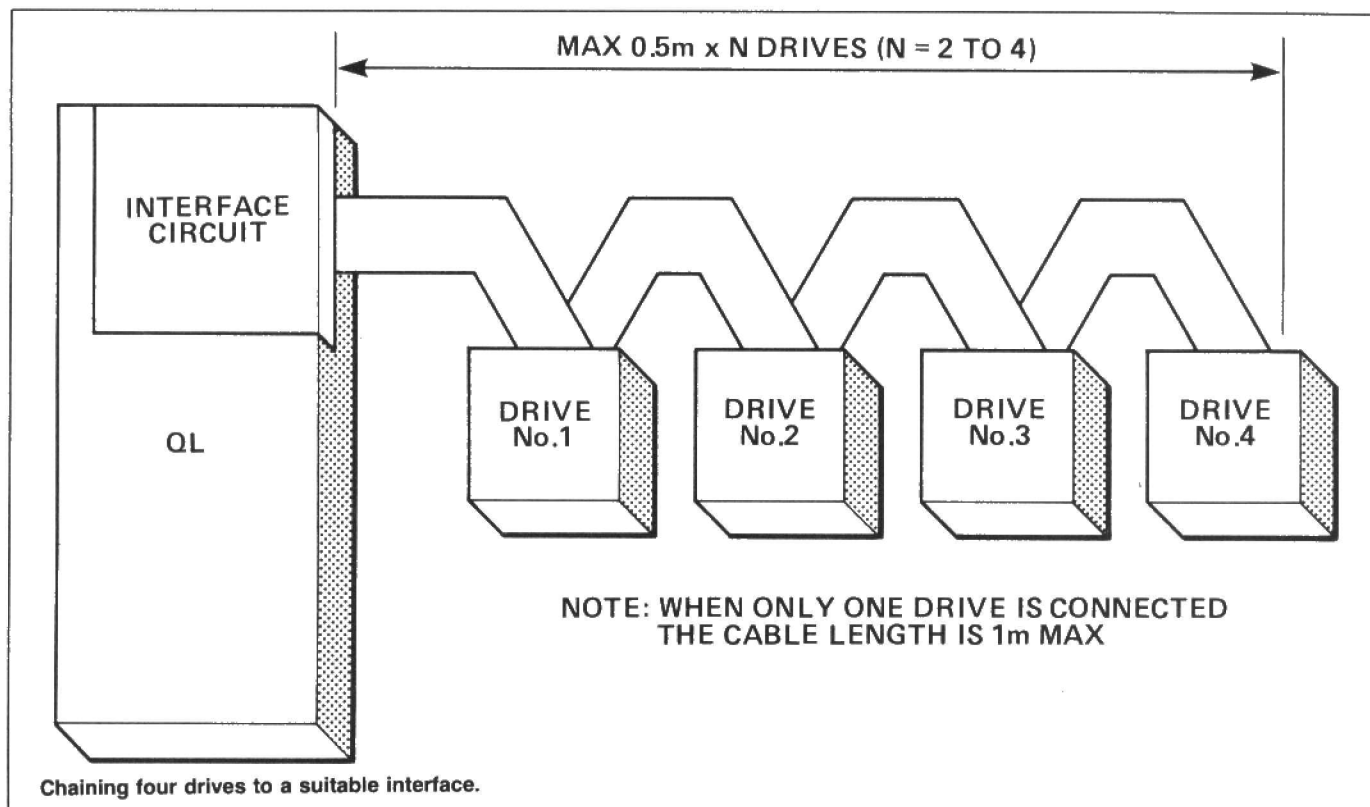
Special motor

The disc is spun in the drive at a constant speed by a rather special type of DC motor. Earlier drives had a different type of motor with a belt drive to the spindle. One or two read/write heads are

lowered on to the disc surface by the electronic or mechanical means. The heads accumulate dirt from the disc, so they need cleaning with a head cleaner disc from time to time.

If you smoke and use the disc storage box as an ash tray, then once-a-week cleaning may be needed. Bear in mind that the cleaner disc is slightly abrasive, so that frequent cleaning will shorten the head life. If some attempt at cleanliness is made, a 10-second clean once a month may be more than adequate.

Finally, I stress that re-alignment of the drive heads is a specialist operation. Do not fiddle with any screws. You will only make matters worse.



THE

P + R : O = G < S

If you have a program worthy of consideration, send it to 'The Progs',
Sinclair QL World, Greencoat House, Francis Street, London SW1P 1DG.
We pay for everything published at the usual page rates.

Program of the month

This program should be particularly useful to students of maths and physics and anyone who needs to visualise mathematical functions. It allows graphs of functions in two-dimensional cartesian (x,y) and polar (r,) coordinates. Functions are entered in standard SuperBasic format and the program takes care automatically of any overflows which can occur in an intelligent manner. Two graphs can be superimposed to solve pairs of simultaneous equations by finding the intersection points and the scale can be adjusted to magnify particular areas of the graph.

The files supplied on Micro-drive are:

boot — autoruns the program;
graph_exe — the program compiled by Supercharge activated using EXE_W;
graph_bas — the basic program file;
gprint_prt — Easel printer dump program;
graph_doc — instruction file in Quill doc form.

By compiling the program an improvement of around 10 times is obtained and so this is necessary to get full benefit from the program.

The available functions and operators are:

*	EXP
*	LN
/	LOG10
+	INT
-	ABS
SIN	RAD
COS	DEG
TAN	SQRT
ASIN	HSIN
ACOS	HCOS
ATAN	HTAN
	PI

where HSIN, HCOS, HTAN are hyperbolic functions and expressions are evaluated in four passes. First functions and constants are evaluated, second ^, third *./ and finally +, -. This can be over-ridden by using parenthesis.

Overflows are detected and can be of three types; overflow — value exceeds $\pm 1E100$; asymptote — value goes to infinity, given by $a/0$ where a is non-zero; not defined — caused by e.g. $0/0$, $SQRT(-1)$, $ASIN(2)$, etc.

The program functions are:

Enter function: The current function is entered in standard

SuperBasic form and the syntax is checked. The maximum length of any expression between brackets is 50 items.

Calculate data. Evaluates the function at the specified number of points. For polar coordinates the range is always 2 while for cartesians the range of x is entered. If the function is to be plotted over the previous function the same range is used.

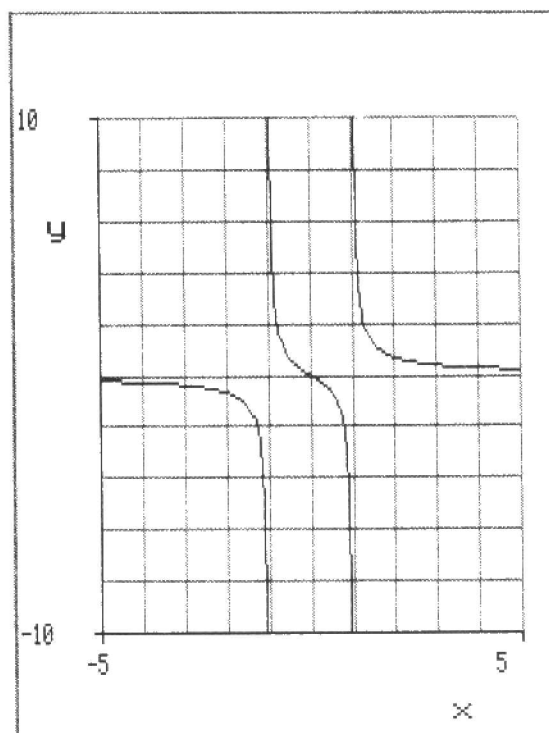
Plot function. Plots the current function and displays any overflows. Either auto-scaling can be used or the range of x and y — or r for polars — can be specified. If the previous function is kept both curves are plotted on the same scale.

Change mode. Switches between polar and cartesian coordinates. Any data is cleared.

Number of points. This allows the number of data-points to be changed in the range 1-1,000. There is a trade-off between the detail of the graph and the time taken to calculate the data. This also clears any data.

Dump to printer. This dumps the screen to an Epson-compatible printer using the Easel printer dump. The dump program is first loaded into memory and is assumed to be on a file on mdv1 called gprint_prt.

Quit program. Returns to SuperBasic.



Mode: CARTESIANS
Number of datapoints=200
Current function:
 $y=x/(x^2-1)$
Range of x:
min=-5
max=5

Range of y:
max=10.2439024
min=-10.2439024
Asymptote at x=-1
Asymptote at x=1

GRAPH PLOTTER by John Banks


```

100 REMark *****
110 REMark * FUNCTION PLOTTER *
120 REMark * John Banks 1987 *
130 REMark *****
140 :
150 initialise
160 set_screen
170 REPEAT main_loop
180   CLS #0:CSIZE #0,2,0
190   PRINT #0,TO 3:UNDER #0,1
200   PRINT #0,'OPTIONS':UNDER #0,0
210   CSIZE #0,1,0
220   PRINT #0,'\ (1) ENTER function'
230   PRINT #0, ' (2) CALCULATE data'
240   PRINT #0, ' (3) PLOT function'
250   PRINT #0, ' (4) CHANGE mode'
260   PRINT #0, ' (5) NUMBER of points'
270   PRINT #0, ' (6) DUMP to printer'
280   PRINT #0, ' (7) QUIT program'
290   key=CODE(INKEY$(-1))
300   CSIZE #0,0,0:CLS #0
310   SELECT ON key
320     =49:get_function
330     =50:calc_data
340     =51:IF polars:plot_polars:ELSE plot_cartes
ians
350     =52:change_mode
360     =53:set_num_points
370     =54:printer_dump
380     =55:EXIT main_loop
390   END SELECT
400 END REPEAT main_loop
410 CLOSE #3:CLOSE #4
420 :
430 DEFINE PROCEDURE set_screen
440 MODE 4:WINDOW 512,256,0,0
450 PAPER 0:CLS
460 WINDOW 300,240,20,0
470 WINDOW #0,170,157,322,83
480 OPEN #3,'scr_472x12a20x241'
490 OPEN #4,'scr_170x82a322x0'
500 BORDER 1,4:INK 7
510 BORDER #0,1,2:PAPER #0,2:INK #0,7
520 BORDER #3,1,4:PAPER #3,4:INK #3,0
530 BORDER #4,1,2:PAPER #4,2:INK #4,7
540 CLS #0:CLS #3:CLS #4
550 CSIZE #3,2,0
560 PRINT #3,TO 6;'GRAPH PLOTTER by John Banks'
570 update_info
580 END DEFINE
590 :
600 DEFINE PROCEDURE update_info
610 CLS #4
620 PRINT #4,'Mode: '
630 IF polars:PRINT #4,'POLARS':ELSE PRINT #4,'CAR
TESIANS'
640 PRINT #4,'Number of datapoints=';num_points
650 PRINT #4,'Current function:'\var2$;'=';fup$
660 IF NOT(polars)
670   PRINT #4,'Range of x:'\min=';min'\max=';max
680 END IF
690 END DEFINE
700 :
710 DEFINE PROCEDURE print_overflows
720 LOCAL i,x
730 PRINT #0,'Range of ';\var2$;'='
740 PRINT #0,'max=';max_val'\min=';min_val
750 FOR i=0 TO num_points
760   x=min+i*(max-min)/num_points
770   IF ABS(y(0,i))=overflow1:PRINT #0,'Overflow
at ';\var1$;'=';x
780   IF y(0,i)=overflow2:PRINT #0,'Undefined valu
e at ';\var1$;'=';x
790   IF ABS(y(0,i))=overflow3:PRINT #0,'Asymptote
at ';\var1$;'=';x
800 END FOR i
810 END DEFINE
820 :
830 DEFINE PROCEDURE plot_cartesians
840 LOCAL y1,y2,x1,x2,xmin,xmax,get_loop
850 print_overflows
860 PRINT #0,'\Auto scaling? (y/n)'
870 IF INKEY$(-1)='y'
880   IF min_val<max_val
890     y1=min_val:y2=max_val
900   ELSE
910     y1=-1:y2=1
920   END IF
930   x1=0:x2=num_points:xmin=min:xmax=max
940 ELSE
950   get_range 'y':y1=range1:y2=range2
960   REPEAT get_loop
970     get_range 'x':xmin=range1:xmax=range2
980     IF xmin>=min AND xmax<=max:EXIT get_loop
990     PRINT #0,er$(2)
1000   END REPEAT get_loop
1010   x1=INT(num_points*(xmin-min)/(max-min))
1020   x2=INT(num_points*(xmax-min)/(max-min))
1030   IF x1=x2:x2=x1+1
1040 END IF
1050 CLS
1060 cart_axes
1070 draw_graph
1080 END DEFINE
1090 :
1100 DEFINE PROCEDURE draw_graph
1110 LOCAL i,j,x_coord,x_prev,x_next,y_coord,val,1
ast_val,coord,draw
1120 FOR i=0 TO dual
1130   IF x1>0:last_val=y(i,x1-1):ELSE last_val=0
1140   val=get_point(y(i,x1),last_val,y(i,x1+1))
1150   IF polars
1160     LINE coord,0
1170   ELSE
1180     LINE -50,coord
1190   END IF
1200   FOR j=x1+1 TO x2
1210     draw=get_point(y(i,j),y(i,j-1),y(i,j+1))
1220     IF polars
1230       x_coord=coord*COS(xmax*j/num_points)
1240       x_prev =coord*COS(xmax*(j-1)/num_points
)
1250       x_next =coord*COS(xmax*(j+1)/num_points
)
1260       y_coord=coord*SIN(xmax*j/num_points)
1270     ELSE
1280       x_coord=100*(j-x1)/(x2-x1)-50
1290       x_prev =100*(j-1-x1)/(x2-x1)-50
1300       x_next =100*(j+1-x1)/(x2-x1)-50
1310       y_coord=coord
1320     END IF
1330     SELECT ON draw
1340       =1:LINE x_coord,y_coord
1350       =2:LINE TO x_coord,y_coord
1360       =3:LINE TO x_prev,-coord,x_next,coord
1370       =4:LINE TO x_prev,coord,x_next,coord
1380     END SELECT
1390   END FOR j
1400 END FOR i
1410 END DEFINE
1420 :
1430 DEFINE FUNCTION get_point(v,vm,vp)
1440 IF v=overflow2:coord=-50:RETURN 0
1450 IF ABS(v)=overflow3
1460   IF vm>=0 AND vp>=0:coord=50:RETURN 4
1470   IF vm<=0 AND vp<=0:coord=-50:RETURN 4
1480   IF vm<0 AND vp>0:coord=50:RETURN 3
1490   IF vm>0 AND vp<0:coord=-50:RETURN 3
1500 END IF
1510 IF v>y2
1520   coord=50
1530   IF vm>=y2 OR vm<=y1:RETURN 1:ELSE RETURN 2
1540 END IF
1550 IF v<y1
1560   coord=-50
1570   IF vm<=y1 OR vm>=y2:RETURN 1:ELSE RETURN 2
1580 END IF
1590 IF polars
1600   coord=50*v/y2
1610 ELSE
1620   coord=50*(2*v-y1-y2)/(y2-y1)
1630 END IF
1640 RETURN 2
1650 END DEFINE
1660 :
1670 DEFINE PROCEDURE cart_axes
1680 LOCAL i
1690 SCALE 140,-70,-70:INK 4
1700 FOR i=-40 TO 50 STEP 10:LINE i,-50 TO i,50
1710 FOR i=-40 TO 50 STEP 10:LINE -50,i TO 50,i
1720 INK 7
1730 LINE -50,-50 TO 50,-50,-50,-50 TO -50,-52,50,
-50 TO 50,-52
1740 LINE -50,-50 TO -50,50,-50,-50 TO -52,-50,-50
,50 TO -52,50
1750 CSIZE 2,0
1760 CURSOR 50,-50,-40,20:PRINT 'x'
1770 CURSOR -50,50,-30,30:PRINT 'y'

```



```

1780 CSIZE 0,0
1790 CURSOR -50,-50,-9,5:PRINT trunc$(xmin)
1800 CURSOR 50,-50,-15,5:PRINT trunc$(xmax)
1810 CURSOR -50,-50,-45,-5:PRINT trunc$(y1)
1820 CURSOR -50,50,-45,-5:PRINT trunc$(y2)
1830 END DEFine
1840 :
1850 DEFine PROCEDURE plot_polars
1860 LOCAL get_loop,xmin,x_max,x1,x2,y1,y2,r$
1870 print_overflows
1880 x1=0:x2=num_points
1890 xmin=0:xmax=2*PI
1900 PRINT #0,'Auto scaling? (y/n)'
1910 IF INKEY$(-1)='y'
1920 IF ABS(max_val)>ABS(min_val):y2=ABS(max_val)
1930 ELSE y2=ABS(min_val)
1940 IF y2=0:y2=1
1950 ELSE y1=-y2
1960 REPEAT get_loop
1970 INPUT #0,'Enter maximum value of r:\r$
1980 IF illegal_num(r$):PRINT #0,er$(1):NEXT g
et_loop
1990 IF r$<>0:EXIT get_loop
2000 PRINT #0,er$(3)
2010 END REPEAT get_loop
2020 y2=ABS(r$):y1=-ABS(r$)
2030 END IF
2040 CLS:polar_axes
2050 draw_graph
2060 END DEFine
2070 :
2080 DEFine PROCEDURE polar_axes
2090 LOCAL i
2100 SCALE 130,-60,-65:INK 4
2110 FOR i=30 TO 330 STEP 30:LINE 0,0 TO 50*COS(RA
D(i)),50*SIN(RAD(i))
2120 FOR i=0 TO 50 STEP 10:CIRCLE 0,0,i
2130 INK 7
2140 LINE 0,0 TO 50,0
2150 CURSOR 50,0,3,-5:PRINT '0=0'
2160 CURSOR 0,50,-15,-15:PRINT '0=pi/2'
2170 CURSOR -50,0,-24,-5:PRINT '0=pi'
2180 CURSOR 0,-50,-15,5:PRINT '0=3pi/2'
2190 CURSOR 50,0,-30,-70:PRINT 'r=';trunc$(y2)
2200 END DEFine
2210 :
2220 DEFine FuNction trunc$(x)
2230 LOCAL i,figs,pt,num$,num2$,ni
2240 i=0:figs=0:pt=0:num$=x
2250 REPEAT loop
2260 i=i+1:IF i>LEN(num$):EXIT loop
2270 ni=CODE(num$(i))
2280 SELECT ON ni
2290 =45:NEXT loop
2300 =46:pt=1:NEXT loop
2310 =69:EXIT loop
2320 END SELECT
2330 figs=figs+1
2340 IF figs>3
2350 IF pt
2360 IF i<LEN(num$):num2$=num$(i+1 TO):ELSE
num2$=''
2370 num$=num$(1 TO i-1)&num2$:i=i-1
2380 ELSE
2390 num$(i)='0'
2400 END IF
2410 END IF
2420 END REPEAT loop
2430 RETURN num$
2440 END DEFine
2450 :
2460 DEFine FuNction sgn(x)
2470 IF x<0:RETURN -1
2480 RETURN 1
2490 END DEFine
2500 :
2510 DEFine PROCEDURE calc_data
2520 LOCAL i,f1$,yi,x
2530 dual=0
2540 PRINT #0,'Keep last function? (y/n)'
2550 IF INKEY$(-1)='y'
2560 dual=1
2570 FOR i=0 TO num_points:y(1,i)=y(0,i)
2580 ELSE
2590 IF polars
2600 min=0:max=2*PI
2610 ELSE
2620 get_range 'x':min=range1:max=range2
2630 END IF
2640 END IF
2650 update_info
2660 CLS #0:PRINT #0,'Calculating'
2670 max_val=-1E100:min_val=1E100
2680 FOR i=0 TO num_points
2690 AT #0,2,0:PRINT #0,'Point: ';i
2700 x=min+i*(max-min)/num_points
2710 f1$=fu$:yi=eval(f1$):y(0,i)=yi
2720 IF ABS(yi)<1E100
2730 IF yi>max_val:max_val=yi
2740 IF yi<min_val:min_val=yi
2750 END IF
2760 END FOR i
2770 END DEFine
2780 :
2790 DEFine PROCEDURE get_range(var$)
2800 LOCAL loop,r1$,r2$
2810 REPEAT loop
2820 PRINT #0,'Enter range of ';var$;':
2830 INPUT #0,r1$;' to ';r2$
2840 IF illegal_num(r1$) OR illegal_num(r2$):PRI
NT #0,er$(1):NEXT loop
2850 range1=r1$:range2=r2$
2860 IF range1<range2:EXIT loop
2870 PRINT #0,er$(2)
2880 END REPEAT loop
2890 END DEFine
2900 :
2910 DEFine PROCEDURE change_mode
2920 PRINT #0,'Data cleared':PAUSE 50
2930 IF polars
2940 polars=0:var1$='x':var2$='y'
2950 ELSE
2960 polars=1:var1$='0':var2$='r'
2970 END IF
2980 DIM y(1,1001)
2990 fu$='0':fup$='0'
3000 min=0:max=1:min_val=0:max_val=0
3010 update_info
3020 END DEFine
3030 :
3040 DEFine PROCEDURE set_num_points
3050 LOCAL loop,n$
3060 REPEAT loop
3070 INPUT #0,'Enter number of datapoints:\n$
3080 IF illegal_num(n$):PRINT #0,er$(1):NEXT loo
p
3090 IF n$>=1 AND n$<=1000 THEN EXIT loop
3100 PRINT #0,er$(3)
3110 END REPEAT loop
3120 num_points=INT(n$)
3130 polars=NOT(polars)
3140 change_mode
3150 END DEFine
3160 :
3170 DEFine PROCEDURE get_function
3180 LOCAL input_loop,check_loop,brackets,item$,sf
lag,x,val,i,j,i1$,i2$,fu2$,f1$,f2$
3190 REPEAT input_loop
3200 PRINT #0,'Enter function:'
3210 IF polars
3220 INPUT #0,'(using x to represent )'\r=';
fu$
3230 ELSE
3240 INPUT #0,'(as function of x)'\y=';fu$
3250 END IF
3260 IF fu$='':NEXT input_loop
3270 brackets=0:error_flag=0
3280 FOR i=1 TO LEN(fu$)
3290 brackets=brackets+(fu$(i)='(')-(fu$(i)=')')
3300 END FOR i
3310 IF brackets>0:PRINT #0,er$(4):NEXT input_l
oop
3320 item$='':sflag=1:i=0:fu$=fu$&'J'
3330 REPEAT check_loop
3340 i=i+1
3350 IF i>LEN(fu$):EXIT check_loop
3360 item$=item$&fu$(i)
3370 i1$=item$(1)
3380 IF i<LEN(fu$):i2$=fu$(i+1):ELSE i2$=''
3390 IF ((i1$>='0' AND i1$<='9') OR (sflag AND
i1$='-')) AND ((i2$<='0' OR i2$>='9') AND i2$<>'.') A
ND item$<>'-' AND NOT(i2$='e')
3400 IF illegal_num(item$):PRINT #0,er$(1):N
EXT input_loop
3410 i=insert_string(' ',i,i+1)
3420 item$='':sflag=0

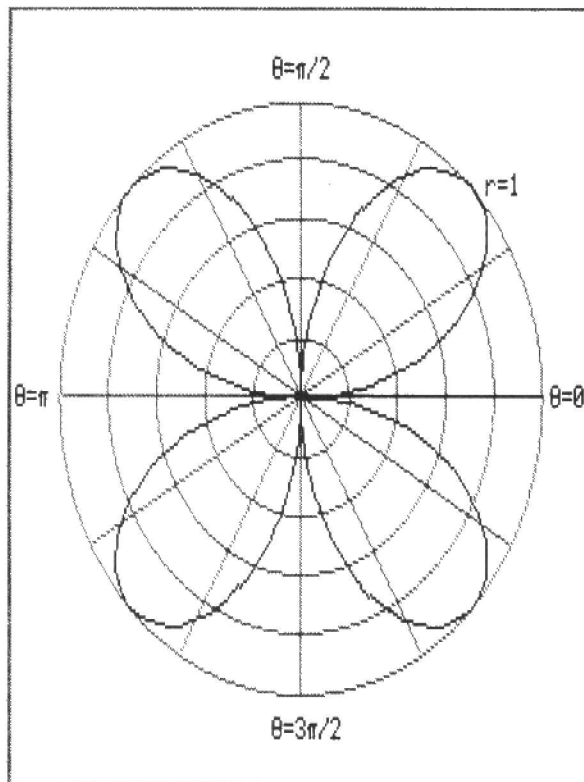
```



```

3430     END IF
3440     FOR j=1 TO 26
3450         IF item$=opp$(j)
3460             IF sflag AND j=6:NEXT check_loop
3470             i=insert_string(CHR$(j+64)&"',",i-LEN(
item$),i+1)
3480             item$=""
3490             sflag=(j=25)
3500         END IF
3510     END FOR j
3520 END REPEAT check_loop
3530 IF item$<>"J":PRINT #0,er$(5):NEXT input_lo
op
3540 fu$=fu$(1 TO i-2)
3550 x=1:fu2$=fu$
3560 val=eval(fu2$)
3570 IF NOT(error_flag):EXIT input_loop
3580 PRINT #0,er$(6)
3590 END REPEAT input_loop
3600 item$="":fup$=""
3610 FOR i=1 TO LEN(fu$)
3620     IF fu$(i)=", '
3630         IF item$>="A" AND item$<="Z"
3640             item$=opp$(CODE(item$)-64)
3650             IF item$="x" AND polars:item$="0"
3660         END IF
3670         fup$=fup$&item$:item$=""
3680     NEXT i
3690 END IF
3700     item$=item$&fu$(i)
3710 END FOR i
3720 update_info
3730 END DEFINE
3740 :
3750 DEFINE FUNCTION insert_string(s$,pos1,pos2)
3760 LOCAL f1$,f2$
3770 f1$="":f2$=""
3780 IF pos1>1:f1$=fu$(1 TO pos1)
3790 IF pos2<=LEN(fu$):f2$=fu$(pos2 TO)
3800 fu$=f1$&s$&f2$
3810 RETURN pos1+LEN(s$)
3820 END DEFINE
3830 :
3840 DEFINE FUNCTION illegal_num(num$)
3850 LOCAL i,pt,loop,ni,n$
3860 n$=num$:IF n$="" OR n$="-":RETURN 1
3870 REPEAT loop
3880     pt=0
3890     IF n$(1)="-":n$=n$(2 TO)
3900     FOR i=1 TO LEN(n$)
3910         ni=CODE(n$(i))
3920         SELECT ON ni
3930             =48 TO 57:NEXT.i
3940             =46:pt=pt+1:IF pt>1:RETURN 1
3950             =69,101:EXIT loop
3960             =REMAINDER:RETURN 1
3970         END SELECT
3980     END FOR i
3990     RETURN 0
4000 END REPEAT loop
4010 IF i=LEN(n$):RETURN 1
4020 n$=n$(i+1 TO)
4030 IF n$(1)="-":n$=n$(2 TO)
4040 FOR i=1 TO LEN(n$)
4050     IF CODE(n$(i))<48 OR CODE(n$(i))>57:RETURN
1
4060 END FOR i
4070 IF n$>99:RETURN 0
4080 RETURN 0
4090 END DEFINE
4100 :
4110 DEFINE FUNCTION eval(f$)
4120 LOCAL i,j,f1$,f2$,f3$,loop,count,ff$
4130 i=1:ff$=f$
4140 REPEAT loop
4150     IF ff$(i)='Y'
4160         count=0
4170         FOR j=i+1 TO LEN(ff$)
4180             IF ff$(j)='Y':count=count+1
4190             IF ff$(j)='Z'
4200                 IF count=0:EXIT j
4210                 count=count-1
4220             END IF
4230         END FOR j
4240         f2$=eval(ff$(i+2 TO j-1))
4250         f1$="":f3$=""
4260         IF i>1:f1$=ff$(1 TO i-1)
4270         IF j<LEN(ff$):f3$=ff$(j+1 TO)
4280         ff$=f1$&f2$&f3$
4290         i=i+LEN(f2$)
4300     END IF
4310     i=i+1
4320     IF i>LEN(ff$):EXIT loop
4330 END REPEAT loop
4340 RETURN calc(ff$)
4350 END DEFINE
4360 :
4370 DEFINE FUNCTION calc(f$)
4380 LOCAL i,j,item,item$,tab%(50),num(50),loop,tt
,nn
4390 i=0
4400 FOR item=1 TO 50
4410     item$=""
4420     REPEAT loop
4430         i=i+1
4440         IF i>LEN(f$):EXIT item
4450         IF f$(i)=", '":EXIT loop
4460         item$=item$&f$(i)
4470     END REPEAT loop
4480     IF item$>="A" AND item$<="Z":
4490         tab%(item)=CODE(item$)-64
4500     ELSE
4510         tab%(item)=-1
4520         num(item)=item$
4530     END IF
4540 END FOR item
4550 IF item=50:error_flag=1
4560 items=item-1
4570 REMARK Pass 1
4580 FOR i=1 TO items
4590     tt=tab%(i)
4600     IF tt=1:num(i)=x:tab%(i)=-1
4610     IF tt=24:num(i)=PI:tab%(i)=-1
4620     IF tt>6 AND tt<24
4630         IF tab%(i+1)<>-1:error_flag=1
4640         nn=num(i+1)
4650         tab%(i)=-1:tab%(i+1)=0
4660         SELECT ON tt
4670             =7:IF ABS(nn)>10000:num(i)=overflow2:E
LSE num(i)=SIN(nn)
4680             =8:IF ABS(nn)>10000:num(i)=overflow2:E
LSE num(i)=COS(nn)
4690             =9:IF ABS(nn)>30000:num(i)=overflow2:E
LSE num(i)=TAN(nn)
4700             =10:IF ABS(nn)>1:num(i)=overflow2:ELSE
num(i)=ASIN(nn)
4710             =11:IF ABS(nn)>1:num(i)=overflow2:ELSE
num(i)=ACOS(nn)
4720             =12:num(i)=ATAN(nn)
4730             =13:IF ABS(nn)<=500
num(i)=EXP(nn)
4740             ELSE
num(i)=EXP(nn)
4750             ELSE
IF nn>0:num(i)=overflow1:ELSE num
(i)=0
4770         END IF
4780         =14:IF nn<=0:num(i)=overflow2:ELSE num(
i)=LN(nn)
4790         =15:IF nn<=0:num(i)=overflow2:ELSE num(
i)=LOG10(nn)
4800         =16:IF nn>1E9:num(i)=nn:ELSE num(i)=INT
(nn)
4810         =17:num(i)=ABS(nn)
4820         =18:num(i)=RAD(nn)
4830         =19:num(i)=DEG(nn)
4840         =20:IF nn<=0:num(i)=overflow2:ELSE num(
i)=SQRT(nn)
4850         =21:IF ABS(nn)>500:num(i)=overflow1*sgn
(nn):ELSE num(i)=(EXP(nn)-EXP(-nn))/2
4860         =22:IF ABS(nn)>500:num(i)=overflow1:ELS
E num(i)=(EXP(nn)+EXP(-nn))/2
4870         =23:IF ABS(nn)>250:num(i)=sgn(nn):ELSE
num(i)=(EXP(2*nn)-1)/(EXP(2*nn)+1)
4880     END SELECT
4890 END IF
4900 END FOR i
4910 kill_zeroes
4920 REMARK Pass 2
4930 FOR i=1 TO items
4940     IF tab%(i)=2:
4950         IF tab%(i-1)<>-1 OR tab%(i+1)<>-1:error_f
lag=1
4960         tab%(i-1)=0:tab%(i)=0:tab%(i+1)=-1
4970         IF ABS(num(i-1))>=1E100:num(i+1)=num(i-1)
:NEXT i
4980         num(i+1)=num(i-1)^num(i+1)
4990     END IF
5000 END FOR i
5010 kill_zeroes

```

Mode: POLARS
Number of datapoints=200
Current function:
 $r = \text{SIN}(2 \times \theta)$

Range of r:
max=1
min=-1

```

5020 REMark Pass 3
5030 FOR i=1 TO items
5040   IF tab%(i)=3 OR tab%(i)=4
5050     IF tab%(i-1)<>-1 OR tab%(i+1)<>-1:error_f
lag=1
5060     tt=tab%(i)
5070     tab%(i-1)=0:tab%(i)=0:tab%(i+1)=-1
5080     IF ABS(num(i-1))>=1E100:num(i+1)=num(i-1)
:NEXT i
5090     SElect ON tt
5100       =3:num(i+1)=num(i-1)*num(i+1)
5110       =4:IF num(i+1)=0
5120         IF num(i-1)=0:num(i+1)=overflow2:E
LSE num(i+1)=overflow3*sgn(num(i-1))
5130       ELSE
5140         num(i+1)=num(i-1)/num(i+1)
5150       END IF
5160     END SElect
5170   END IF
5180 END FOR i
5190 kill_zeroes
5200 REMark Pass 4
5210 FOR i=1 TO items
5220   IF tab%(i)=5 OR tab%(i)=6
5230     IF tab%(i-1)<>-1 OR tab%(i+1)<>-1:error_f
lag=1
5240     tt=tab%(i)
5250     tab%(i-1)=0:tab%(i)=0
5260     IF ABS(num(i-1))>=1E100:num(i+1)=num(i-1)
:NEXT i
5270     SElect ON tt
5280       =5:num(i+1)=num(i-1)+num(i+1)
5290       =6:num(i+1)=num(i-1)-num(i+1)
5300     END SElect
5310   END IF
5320 END FOR i
5330 kill_zeroes
5340 IF items<>1:error_flg=1
5350 IF ABS(num(1))>=1E100 AND ABS(num(1))<>overfl
ow1 AND ABS(num(1))<>overflow3 AND num(1)<>overflo
w2:num(1)=overflow1*sgn(num(1))
5360 RETURN num(1)
5370 END DEFINE
5380 :
5390 DEFINE PROCEDURE kill_zeroes
5400 LOCAL i,j
5410 j=1
5420 FOR i=1 TO items
5430   IF tab%(i)<>0
5440     tab%(j)=tab%(i)
5450     num(j)=num(i):j=j+1

```

```

5460   END IF
5470 END FOR i
5480 items=j-1
5490 END DEFINE
5500 :
5510 DEFINE PROCEDURE initialise
5520 LOCAL i,e$
5530 num_points=200:polars=0
5540 DIM opp$(26,5),y(1,1001),er$(6,30)
5550 RESTORE 5670
5560 FOR i=1 TO 26:READ opp$(i)
5570 FOR i=1 TO 6:READ e$:er$(i)='ERROR: '&e$
5580 overflow1=1E500:overflow2=1E510:overflow3=1E5
20
5590 error_flag=0
5600 fu$='0':fup$='0':overflow_flag=0
5610 max=1:min=0:max_val=0:min_val=0
5620 dual=0:var1$='x':var2$='y'
5630 dump_addr=0
5640 END DEFINE
5650 :
5660 REMark Operators
5670 DATA 'x','^','*','/','+','-
5680 DATA 'SIN','COS','TAN','ASIN','ACOS','ATAN'
5690 DATA 'EXP','LN','LOG10','INT','ABS','RAD'
5700 DATA 'DEG','SQRT','HSIN','HCOS','HTAN'
5710 DATA 'PI','(',')'
5720 REMark Error messages
5730 DATA 'Invalid number'
5740 DATA 'Invalid range'
5750 DATA 'Out of range'
5760 DATA 'Brackets dont match'
5770 DATA 'Invalid function'
5780 DATA 'Syntax incorrect'
5790 :
5800 DEFINE PROCEDURE printer_dump
5810 IF dump_addr=0
5820   PRINT #0,'Load printer dump? (y/n)'
5830   IF INKEY$(-1)='y'
5840     dump_addr=RESPR(2000)
5850     PRINT #0,'\Insert cartridge in mdv1\'and
press a key'
5860     PAUSE
5870     LBYTES 'mdv1_gprint_prt',dump_addr
5880   ELSE
5890     RETURN
5900   END IF
5910 END IF
5920 CLS #0:print_overflows
5930 CALL dump_addr
5940 END DEFINE

```




86 Commercial Road, Paddock Wood, Kent TN12 6DT

089283 2552

COMBINED MDV T'KIT £19.95

MDV and ALMDV toolkits on one microdrive

FORTH £29.95

Professional Forth 83 compiler
● Fast and compact ● Many extensions

Computer One Professional Monitor — NEW — £39.95

(Upgrade price £20.00 if you return old cartridge — You need to keep the old manual)

New features:

- Symbolic debugging including symbol management
- Integral 88000 assembler for single/multi-line assembly from files.
- Include files.
- Dual screen debugging.
- Function key control of monitor clones.
- Function key control of monitor clones.
- New commands: ass, noass, -, nojb, include, noinclude, nosymbols, noeval, swap, noswap, nodlines, noflags, error, escape, sleep, base, "
- All of the above in addition to:
- Integral disassembler
- Tracing of code in RAM or ROM
- Move and search memory
- Queue tracing
- Memory modify and display
- Breakpoints in RAM or ROM
- Full job inspection and control
- Configurable multi-window display

ICE

LOOK FOR SPECIAL OFFERS

ICE — £26.95 + Choice Multitasking Program £14.95 + Toolkit especially for ICE users £14.95.

Limited while stocks last

ICE is a ROM based utility program that turns your QL with an ICON controlled "state of the art" computer. Not only does it add a number of functions not found in a standard QL including calculator and calendar, it also makes all basic functions a dream to use.

ICE + MOUSE

Was £79.95

NOW £59.95p The Eidersoft mouse combined with the above ICE ROM gives your QL the power of the Apple Macintosh or GEM Mouse control is implemented on all packages that use the ICE system, which gives a very smooth and fast response that will not be out of sync if you already have ICE. We offer £10.00 trade-in if you return your existing ICE.

ARTICE OR MOUSEART

£14.95p

A keyboard or mouse controlled graphics program. The features include circle, line box, freehand draw, paintbox colours and brushes, copy transfer and save image. Epson printer dump, undo, spray can, recolour, text, mode 4 and 8 operation.

ICICLE

£14.95p

A multi-tasking utility that allows you to set up your own ICONS for the control of programs, including QUILL, ABACUS and a host of other programs. As well as a full ICON editor it includes a printer spooler and printer manager.

DRAWING OFF-ICE

£24.95p

3D screen designer aimed at producing true 3D images on your screen. The perspective is automatically calculated to a given vanishing point. This is not some complex co-ordinated program which requires you to work out every point of the 3D object in advance. Requires 3 1/2" disk drive. This program now includes version of mouse art. Expanded memory required.

NEW PUBLISHING OFF-ICE £24.95p

Incorporating an enhanced version of mouse art. Everything you need to design your newspaper or publication. Disk + Expanded Memory only required.

GAMES

JUST BACK!! *SCRABBLE

A great version of an old favourite. One to four players, 8 skill levels, 12,000 word dictionary.

Only £12.50

Summer Reduction. Limited time only

BJ RETURNS

£10.95p

Is the excellent sequel to QL Caverns incorporating many of the excellent features and graphics of the original game.

BJ IN 3D LAND

£10.95p

BJ is lost again! This time in a 3D maze that will pose a challenge to all 70 screens.

*KARATE

Was £19.95 NOW £12.50

Incredible graphics. 18 different movements including kicks, punches and somersaults. Multi screen action, large flicker free sprites, multi player option, sound effects, keyboard or joystick.

SPOOK

£10.95p

Probably one of the best versions of Pacman for any Micro computer.

GAMES PACK

£16.95p

Comprising of ZAPPER, the classic arcade game. Eagle a defender type game and Citadel where you must unravel the secrets of a giant city as you steer your craft through 50 screens of danger and excitement and, finally, BJ returns.

NEW!

COMPUWARE

1 TO 1 DUMP £5.99

Undistorted screen dump to an Epson FX80 printer

ALMDV TOOLKIT £14.95

1000 lines of commented source code for mdv header and sector access

ORGANISER The machine that thinks with you

From **£99.00**

BANK MANAGER ALARM CLOCK DIARY MATHEMATICIAN

Diary
Alarm
Calculator
Address Book
Organiser II CM £99.00p
Organiser II XP £139.50p
Finance Manager £29.95p
Dictionary £29.95p
Comms. link £59.95p
Spreadsheet £39.95p
6K datapak £19.95p
32K datapak £34.95p
64K datapak £79.95p
128K datapak £99.95p
Book on Organiser £10.95p
QL to Organiser £39.95p
Leather case for Organiser, battery and datapak £12.95
Nylon waterproof protective bag £10.95p
Holster £39.95p
Real leather organiser fax. Holds organiser, tax paper and pens £69.95p

OBASE
A database creator, allows you to create your own database, 6 different field types. Import, export to archive, use main database supplied on 32K datapak. Not copy protected. Plus free calculator to calculate % and %.

£69.95
*Buy a leather fax case and save 10% on any other Organiser purchase

BOOT 128K £5.99

Run "128K only" programs in your expanded QL

QL PASCAL £29.95

Exceeds the ISO standard!!
● Fast ● Efficient
● Simple to use ● Professional

TYPING TUTOR £14.95

Professional tuition in easy steps. ● Speed and accuracy measurement.

PRO-MONITOR £29.95

As monitor plus symbolic debugging (return old cart. + £15 to upgrade).

ASSEMBLER £19.95

The fastest QL Assembler.
● Editor ● Linker

MULTI-PRINT £5.99

Print multiple copies of files with this multitasking program

MONITOR £15.95

Powerful full featured debugger
● Many unique facilities.

COPYCAT £10.99

Backs up most protected cartridges in no time.

MDV TOOLKIT £9.99

Extensions to read/write mdv sectors, headers, etc.

TASK SWOPPER (Version 2) NEW £19.95

(Upgrade price £10.00 if you return your old Task Swopper manual)

New features:

- True job cloning. (Saves memory, eg: 9 Psion clones in a 640K QL leaves 400K free!)
 - Ready made and easily customised start-up menu program.
 - Automated printer driver selection (useful for program swapping and also for selecting different typesets from your printer).
 - Automated set up of the QL clock minimises typing by remembering the year/month/day.
 - Multitasking clock program.
 - Compatible with the WL front end program QATS.
- All these features, and Task Swopper still only uses 10K of RAM, and is compatible with floppy discs, ROM toolkits, RAM discs and useful utilities such as QL Keydefine.

MEGA TOOLBOX — NEW — £29.95 (Demonstration cartridge — Redeemable against purchase: £10.00)

This is not just another run-of-the-mill QL toolkit. It adds over 168 new commands to QL basic, and truly breaks new ground. extensive and original use is made of the QL's multi-tasking ability, enabling the basic programmer to perform wonders, even if he wants to compile his programs.

Main Features:

- Designed for use by serious programmers and software houses.
 - Improved control of QDOS resources (including memory, keyboard, pipes, files, jobs, alarm clocks and tune playing jobs).
 - Windows/graphics (saving, restoring, copying, mirroring — optional compression).
 - New keyboard input driver for better command line editing.
 - Drawing/text printing commands (eg 3D text), ideal for constructing animated slide-shows for games/advertising, etc.
 - Dual screen handling (copying, swapping, automated screen mode control).
- Here are just 39 of the 168 new commands: FREE_MEM, ALCP RCHP, MCOPI, MFILE, MSEARCH, FACT, DAYS, UPPERS, LOWERS, HEX, BIN, DEC, FILE_LEN, PPOS, GET, PUT, KEYBOARD, ENTER, ACTIVATE_O, STICK PROMPTS, JOBS, JOB_STAT, REMOVE, PIPE_ID, CONNECT, SET_FONT, PRINT_3D, MPRINT, EXPAND, HIDE, SHOW, ZOOM, PRINT_X, SLIDE_X, ALARM_X, QTRAP, OCALL.

EXPERT SYSTEM SHELL NEW — £49.95

This is a serious tool designed both to introduce the novice to the design of expert systems, and to be used for serious expert system work. The expert system programmer can construct sophisticated rule based systems and put them into real applications. A tutorial will help you learn how to design an expert system and for ease of use, context sensitive help is provided.

Main Features:

- Tokenised rules save money (about 100 rules in a 128K QL).
- Precompiled expressions for fast rule evaluation.
- Intelligent searching and precompiling of rule base.
- Boolean and fuzzy logic plus user definable probability relationships.
- Forging of input and output and ability to construct menus for the user.
- Original constructs: FIRST OF, MIN OF, MAX OF, ALL OF
- Conditional operators: =, >, <, <=, >=, <=, >=, <=, >=
- Multiple goal paths ... and much more.

UTILITIES

GRAPHICS CONSTRUCTION KIT £14.95p

A set of utilities that allows you to construct your own pulldown menus in SUPERBASIC.

Q-SWITCH

£14.95p

Multi-tasking that permits a quick switch from ARCHIVE to ABACUS QUILL or EASEL, suspending tasks while jumping to another task. Written by the author of ICICLE and KEYDEFINE this is a very popular package.

QSPELL

£24.95p

A spelling checker for Quill, supplied with a dictionary of 250,000 words and checks an A4 page in 24 secs.

Q-FLASH RAM DISK

£14.95p

Allows the user to define RAM disc, programs can then be instantly saved and re-loaded. A print spooler is also included to allow you to print whilst using the programs.

QL to PC

£29.95p

Transfer your existing files to a PC for use within XCHANGE, PC FOUR or any standard DATABASE SPREADSHEET or WORDPROCESSOR. Price includes software for both machines and cable.

QL TO ORGANISER

£39.95p

Transfer both ways your existing QL files within exchange. Complete with software and cable.

QL TECHNICAL BOOKS (Set of 7)

*Special price to clear £12.95p

PC FOUR for IBM compatibles

Our price £69.00p ex VAT

Microdrive Storage Box £4.95p
Microdrive Storage Box with 10 cartridges £22.45p
Microdrive Storage Box with 20 cartridges £39.95p

WE SELL MOST QL PRODUCTS — PHONE FOR DETAILS

AMAZING

SPRING
OFFER

FREE

TRANSFORM STORAGE
BOX WITH FIRST 100
ORDERS. VALUE £495.00